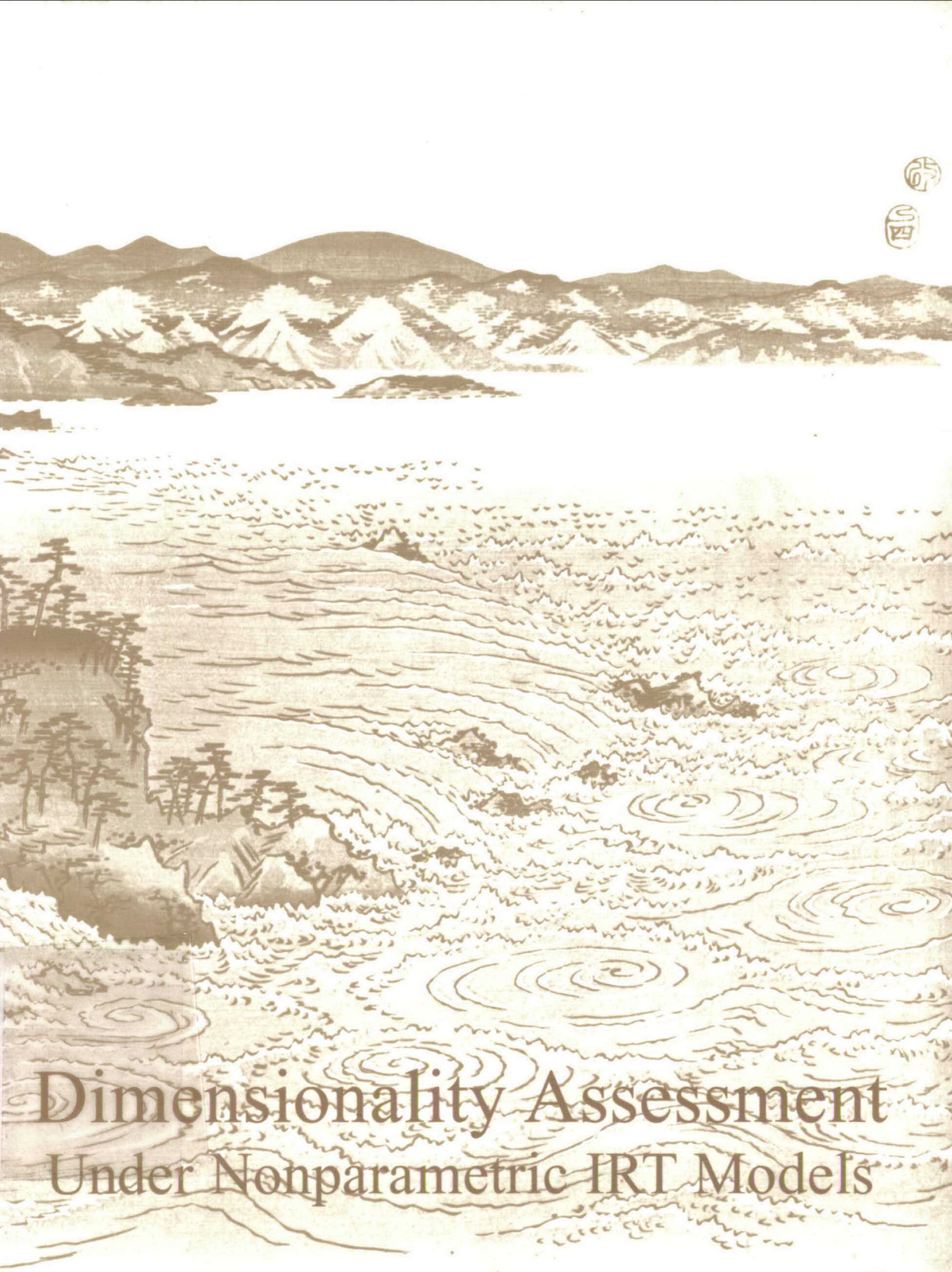
Dimensionality assessment under nonparametric IRT models

van Abswoude, A.A.H.

Publication date:
2004

Document Version
Publisher's PDF, also known as Version of record



Dimensionality Assessment
Under Nonparametric IRT Models

Alexandra A. H. van Abswoude

Uitnodiging

voor het bijwonen van
de verdediging
van mijn proefschrift

Dimensionality Assessment Under Nonparametric IRT models

op 14 mei 2004
om 14:15
in de aula van de
Universiteit van Tilburg
Warandelaan 2
Tilburg

en de
receptie na afloop

Sandra van Abswoude
020 - 421 3824
AAHvanAbswoude
@Yahoo.com

Alexandra A. H. van Abswoude

Dimensionality Assessment Under Nonparametric IRT Models

ISBN 90-9018047-8
Printed by PrintPartners Ipskamp, Enschede
Cover illustration: Andō Hiroshige (1797-1858). View of the whirlpool at Naruto. Digital material by courtesy of Hotei Japanese Prints/Ukiyo-e Books, Leiden, The Netherlands

Dimensionality Assessment Under Nonparametric IRT Models

(Dimensionaliteitsonderzoek Onder Niet-Parametrische IRT Modellen)

Proefschrift

ter verkrijging van de graad van doctor aan de Universiteit van Tilburg,
op gezag van de rector magnificus, prof. dr. F. A. van der Duyn Schouten, in het
openbaar te verdedigen ten overstaan van een door het college voor promoties
aangewezen commissie in de aula van de Universiteit
op vrijdag 14 mei 2004 om 14.15 uur

door

Alexandra Alida Hendrika van Abswoude

geboren op 24 september 1972 te Oegstgeest

Promotores: Prof. dr. K. Sijtsma
Prof. dr. J. K. Vermunt
Copromotor: Dr. B. T. Hemker

Acknowledgements

For teaching me the tricks of the trade, creating a stimulating environment, giving valuable criticism, or being supportive in the last four years, I would like to thank: (De volgende mensen wil ik graag bedanken omdat zij me in de afgelopen vier jaar stimuleerden, met me meedachten, of steunden:)

supervisors Klaas Sijtsma, Jeroen Vermunt and Bas Hemker; members of the 'Ordinal Measurement' research group Andries van der Ark, Wilco Emons, Dave Hessen, Don Mellenbergh, Ivo Molenaar and Marieke van Onna; Bill Stout and his lab members at the Department of Statistics at the University of Illinois; colleagues from IOPS and WORC; my colleagues at the Department of Methodology and Statistics of Tilburg University, especially Emmanuel Aris, Marcel van Assen, Wicher Bergsma, Samantha Bouwmeester, Liesbet van Dijk, Francisca Galindo Garre, John Gelissen, Joost van Ginkel, Janneke te Marvelde, and Marieke Spreeuwenberg; former Ph.D. students at the Department of Psychology Seger Breugelmans and Marloes van Engen; Ph.D. students at Methodenleer of the University of Amsterdam; my friends, especially Mui Sian Liauw (Anyo), Romke Rouw and Merlijn Wouters; Karin Hendriks and my dear brother Japhet van Abswoude; and my parents Jan van Abswoude and Anneke van den Dool.

Thank you all! (Iedereen bedankt!)

Amsterdam, March 2004,
Sandra van Abswoude

Contents

Introduction

Tests and questionnaires provide scientists and practitioners from various disciplines like psychology, educational science and political science with objective means to measure subjects with respect to their traits, abilities, or attitudes. Such measurement can be relevant in many research settings such as the selection or placement of students in certain school types, the diagnosis for psychological or medical treatment, or the selection of the best applicants for a job.

Tests may be aimed at measuring one or multiple abilities. A test aimed at measuring one ability like a mathematics skill may, however, be sensitive to other sources of variation as well. The subjects' test scores need not be the same every time a test is taken because the test circumstances need not be the same (e.g., noisy surroundings, or having had a party the night before). Standardized testing practices as discussed in textbooks on research methodology (e.g., Cronbach, 1990) will control for most situational factors. Also, the topic or the wording of one or two mathematics problems (items) may unintentionally draw on other abilities and, as a consequence, may give one group of subjects a advantage over another. For example, an item involving a baseball court may give children from the USA an advantage over European children. The effects of these "nuisance" factors on the subject's test performance may cancel each other out when the number of items is large (e.g., Stout, 2002). Tests of this type are driven by one "dominant" ability.

Tests may also measure multiple abilities. For example, test items may draw upon the students' language skills as well as on their mathematics skills. This may occur in contextual math problems. For subjects with equal language skills, this will not cause extra variation in test scores and, thus, the test is driven by one dominant ability. When subjects have different language skills, this will cause extra variation in the test scores. Students with poor language skills (e.g., dyslexia, English not being their first language) may perform worse on this test than one would expect based on their mathematics ability alone. Ignoring language as a

source of variation may lead to seriously unjust decisions for these students. Data that result from the confrontation of subjects to these test items comprise multiple abilities but none of them is dominant.

Alternatively, a test may be sensitive to multiple abilities, but each test item is driven by one dominant ability. An example is a mathematics test that targets different sub-abilities like spatial insight, arithmetics, and calculus. These sub-abilities may be related to each other. Another example is an intelligence test that targets different sub-abilities like verbal reasoning, quantitative reasoning and abstract/visual reasoning (e.g., the Stanford-Binet intelligence scale; see Thorndike, Hagen, & Sattler, 1986). Data resulting from a test measuring these sub-abilities may exhibit "approximate simple structure" (e.g., Stout, 1987). Simple structure in practice does not occur because unintended factors will to some extent influence the subjects' responses. One may note that data with one dominant ability also reflect approximate simple structure. For approximate simple structure data it is possible to partition the total test into sub-tests driven by one dominant ability. This is convenient because measuring subjects is mathematically and conceptually much easier when based on a single ability. This thesis discusses methods that can be used to select one or more sets of items, each driven by one dominant ability, from a test measuring multiple abilities.

The traits, abilities, and attitudes that social scientists try to observe using tests are inherently unobservable in nature. In item response theory (IRT; e.g., Mokken, 1971; Hambleton & Swaminathan, 1985; Fischer & Molenaar, 1995) they are for that reason called "latent traits". The term "dimensionality" refers to the number of latent traits that can explain the responses of subjects to a set of items or a test. A set of items that is driven by a single latent trait is denoted "unidimensional" and by multiple latent traits "multidimensional". IRT provides a statistical theory that defines the relationship between the latent traits and the probability that the subject gives a particular response on an item. The function that defines this relationship is denoted an item response function. As the number of parameters that defines an IRT model decreases, the model becomes easier to estimate and the measurement properties that apply under the model become more attractive. Under the one-parameter logistic model (Rasch, 1960) for example, measurement of abilities on an interval level is possible (i.e., concerning three students named Max, Sien and Bobby measured on a logit scale who have latent trait scores 0.5, 1 and 2, we can say that the difference in ability between Bobby and Sien was twice as large as between Max and Sien). A trade-off when using few parameters is, however, that it is less likely that the model gives a good representation of the data.

Nonparametric IRT models are based on the same assumptions as parametric IRT models (i.e., unidimensionality, local independence and monotonicity), but the item response functions in these models are not parametrically defined (see Stout, 2002; Sijtsma & Molenaar, 2002 for an overview). These properties make nonparametric IRT models appropriate for the ordering of subjects and, for a particular model, of items. The ordinal nature implies that compared to their parametric counterparts weaker statements can be made about the subjects (i.e., we may infer that Bobby's mathematical ability was better than Sien's ability, and that Sien's was better than Max's ability, but not how much better). The advantage lies in the fact that nonparametric models will more likely fit data than parametric models.

When selecting items into one or more approximately unidimensional sets (scales) within the framework of nonparametric IRT, different approaches can be used. Mokken Scale analysis for Polytomous items (MSP; e.g., Molenaar & Sijtsma, 2000) focusses on the monotonicity assumption of IRT models by using a scaling coefficient (H coefficient; Loevinger, 1948; Mokken, 1971) that is sensitive to the discriminations of items. The use of this coefficient makes the method insensitive to the distribution of the difficulty of the items because it corrects for the items' marginal distributions. Another attractive feature is that the user can choose a suitable lower bound for item and scale quality. Hemker, Sijtsma, and Molenaar (1995) demonstrated that these scales generally reflect the underlying dimensionality of data, but the scales can hold a few items sensitive to a different latent trait than the remainder of the items in a scale. The methods DETECT, DIMTEST and HCA/CCPROX (e.g., Stout, 2002, for an overview) use a relaxation of the local independence assumption of IRT models. These methods seem to aim more directly at obtaining unidimensional subsets.

Organization of the Chapters

This thesis presents some contributions to dimensionality assessment under nonparametric IRT models. The following research questions can be distinguished in this thesis: (a) How successful is the scaling method MSP compared to the dimensionality assessment methods DETECT, DIMTEST and HCA/CCPROX?, (b) Why does MSP sometimes select an item into a scale that is driven by a different trait than the other items in the same scale; is the cause the scaling coefficient, the algorithm, the side conditions, or a combination of these?, (c) How can MSP be improved such that unidimensional scales may be obtained and the attractive properties of the current method are maintained?

Chapter 1 covers the first research question. It discusses two models on which dimensionality assessment methods in nonparametric IRT can be based: the essentially and the strictly unidimensional models. These models are compared theoretically. Using a simulation study, three essentially unidimensional model based methods DETECT, DIMTEST and HCA/CCPROX and one strictly unidimensional model based method, MSP, are compared on their ability to assess the dimensionality of different types of data. Recommendations are given when to use which method.

Chapters 2 through 5 aim to answer the last two research questions. In Chapter 2, four hierarchical alternatives for the item selection algorithm used for Mokken Scale Analysis are proposed. Attractive properties of these algorithms are their simplicity, their availability in standard software packages for the social sciences like SPSS, and the opportunity they provide to investigate the process by which sets of items are joined. By means of a simulation study and an empirical example, the success of these hierarchical methods in assessing dimensionality is compared with respect to each other and to MSP's item selection method.

The third chapter discusses the effects that different clustering algorithms may have on finding the underlying dimensionality of data. Using a few examples, we illustrate where in the process of clustering things might go wrong in the sense that suboptimal solutions may be found and, consequently, the underlying dimensionality cannot be retrieved.

The next chapter, Chapter 4, introduces three alternative methods aimed at reducing the probability of obtaining suboptimal solutions. These methods use deterministic and stochastic versions of non-hierarchical clustering algorithms and clearly defined scaling objectives in both unidimensional and multidimensional contexts. Specific scaling conditions are not included. Using a simulation study, we investigate whether stochastic algorithms may be used for obtaining optimal (or, nearly optimal) solutions. Moreover, we investigate how successful these stochastic methods based on the H coefficient are in yielding sets that reflect the underlying dimensionality of data.

Finally, in Chapter 5, suggestions are presented on how the new stochastic methods of Chapter 4 may be extended so that they become useful for creating multiple Mokken scales; that is, incorporating the Mokken scale analysis conditions. The chapter also explains how other interesting conditions may be imposed on the data as well.

Chapter 1

Comparing Dimensionality Assessment Procedures Under Nonparametric IRT Models

Abstract

In this chapter four methods for dimensionality assessment under nonparametric item response theory methods (MSP, DETECT, HCA/CCPROX, and DIMTEST) were compared. First, the methods were compared theoretically. Second, a simulation study was done to compare the effectiveness of MSP, DETECT, and HCA/CCPROX in finding a simulated dimensional structure of a matrix of item response data. In several design cells, the methods that use covariances conditional on the latent trait (DETECT and HCA/CCPROX) were superior in finding the simulated structure to the method that used normed unconditional covariances (MSP). Third, the correctness of the decision of accepting or rejecting unidimensionality based on the statistics used in DETECT and DIMTEST was considered. This decision did not always reflect the true dimensionality of the item pool.

This chapter has been published as: Van Abswoude, A.A.H., Van der Ark, L.A. & Sijtsma, K. (2003). A comparative study on test dimensionality procedures under nonparametric IRT models. *Applied Psychological Measurement*, 28 (1), 3-24.

1.1 Introduction

Although it can be argued that test performance often is simultaneously governed by several latent traits, most researchers seem to agree that a test or a questionnaire should preferably measure only one dominant latent trait. This is reflected by the existence of many unidimensional item response theory (IRT) models and only a few multidimensional IRT models (e.g., Kelderman & Rijkes, 1994; Reckase, 1997). There are at least two reasons why unidimensional measurement is preferred.

First, when test data measure one latent trait, a single score can be assigned to each examinee, and the interpretation of test performance is unambiguous. Also, when a measurement practitioner intends to measure multiple latent traits, it can be argued that he/she should construct a unidimensional test for each trait separately. When items measuring different traits are part of the same test, for example, when some items are sensitive to vocabulary and others are sensitive to verbal comprehension, this line of reasoning would stipulate that the test is split into two unidimensional subtests, and that examinees obtain separate scores on each. Note that if one summary score would be assigned based on both item types, it would be unclear to what degree a latent trait influenced the test score of a particular examinee, because one ability could have compensated for the other, also depending on the strength of their mutual relationship.

Second, due to the larger number of parameters the estimation of multidimensional IRT models is more complicated than the estimation of unidimensional IRT models (e.g., see Béguin & Glas, 2001, who used Markov chain Monte Carlo techniques for estimating a multidimensional normal ogive model). Using the simpler unidimensional IRT models instead may be an attractive option, in particular, after an item clustering method has been applied to the data to determine their dimensionality. Then, a unidimensional IRT model can be fitted to the items loading on a particular latent trait, and this may be repeated for each latent trait.

Traditionally, the dimensionality of responses from a set of dichotomous items was determined using linear factor analysis. It is well known that 'difficulty factors' may arise (Hattie, Krakowski, Rogers, & Swaminathan, 1996; Nandakumar & Stout, 1993; see Miecskowski et al., 1993, for an example) when items vary widely in difficulty, and correlations are based on binary item scores. Other problems may arise when tetrachoric correlations are used to correct for the extreme discreteness of the binary item scores. One problem is that the tetrachoric correlation matrix may not be positive definite (Knol & Berger, 1991; Lord & Novick, 1968, p. 349). Another problem is that tetrachoric correlations estimate a correlation based on

hypothesized normal variables when, in fact, only binary scores were observed, and normality thus may be an invalid assumption. An alternative may be nonlinear factor analysis, but Hattie et al. (1996) found that nonlinear factor models were not as effective in discriminating between unidimensional and multidimensional data sets as their linear counterparts.

An alternative to factor analysis is nonparametric item response theory (NIRT), which is central in this chapter. NIRT uses a nonlinear model for the relation between binary correct/incorrect item scores and a continuous latent trait, and has the advantage that it can be applied directly to the binary item scores. This means that tetrachoric correlations are not necessary. The purpose of this study was to investigate the effectiveness of three methods used for retrieving the dimensionality of binary item score data, which are based on NIRT and which use covariances between binary item scores. We consider the methods as they exist 'of the shelf'. The three methods considered here were MSP (Hemker et al., 1995; Molenaar & Sijtsma, 2000), DETECT (Kim, 1994; Zhang & Stout, 1999a, 1999b), and HCA/CCPROX (Roussos, 1992; Roussos, Stout, & Marden, 1998). In addition, the statistical procedure DIMTEST (Nandakumar & Stout, 1993; Stout, 1987; Stout, Douglas, Junker, & Roussos, 1993; Stout, Goodwin Froelich, & Gao, 2001) was used for testing hypotheses about the dimensionality of item response data, and results were compared to the results of the other methods.

1.2 Nonparametric IRT

1.2.1 Strictly and Essentially Unidimensional Models

Strictly unidimensional models. Let $\mathbf{X} = (X_1, \ldots, X_J)$ be the vector of J binary scored item variables, and let $\mathbf{x} = (x_1, \ldots, x_J)$ be the realization of $\mathbf{X}$. Score 1 indicates a correct answer, and score 0 an incorrect answer. The probability of an item score of 1 depends on one latent trait θ, and is denoted $P_j(\theta)$. This is the unidimensionality (UD) assumption. Probability $P_j(\theta)$ is the item response function (IRF). Further, local independence (LI) is assumed, which is defined as

$$P(\mathbf{X} = \mathbf{x}|\theta) = \prod_{j=1}^{J} P(X_j = x_j|\theta). \tag{1.1}$$

Assumption LI means that given a fixed value of θ the responses of an individual to the J items are statistically independent. Assumptions UD and LI together do not imply falsifiable consequences on the observed data (Holland & Rosenbaum, 1986; Junker, 1993). For this purpose, we need restrictions on the IRFs. For

example, let θ_a and θ_b be the latent trait values of examinees a and b, then the monotonicity assumption (M) states that,

$$P_j(\theta_a) \leq P_j(\theta_b), \text{ whenever } \theta_a < \theta_b, \text{ for } j = 1, \ldots, J.$$

Assumption M means that the IRFs are monotone nondecreasing in θ. The assumptions of UD, LI and M together define the model of monotone homogeneity (Mokken & Lewis, 1982; Sijtsma & Molenaar, 2002, chap. 2-5). The model of monotone homogeneity is an NIRT model that implies the stochastic ordering of θ by the total test score, $X_+ = \Sigma X_j$ (Grayson, 1988; Hemker, Sijtsma, Molenaar, & Junker, 1997). A more restrictive model can be defined by adding to UD, LI, and M the assumption that the IRFs do not intersect. Together these four assumptions define the model of double monotonicity (Mokken & Lewis, 1982; Sijtsma & Molenaar, 2002, chap. 2, p. 6). In addition to ordinal person measurement the model of double monotonicity allows an invariant item ordering (Sijtsma & Junker, 1996).

Essentially unidimensional models. Stout (1990; also, see Junker, 1993) defined the dimensionality of item response data in terms of the minimum number of traits necessary to achieve LI and M. In essentially unidimensional models, however, the assumptions of LI and M are relaxed to essential independence and weak monotonicity, respectively. Stout (1990) assumed that test performance is governed by a dominant latent trait and several nuisance latent traits. Following this idea, a vector $\boldsymbol{\theta} = (\theta, \theta_1, \ldots, \theta_W)$ represents the dominant θ and W nuisance traits. Based on large sample theory, *essential independence* (EI; Stout, 1990) states that,

$$\binom{J}{2}^{-1} \sum_{1 \leq j < k \leq J} |\text{Cov}(X_j, X_k | \boldsymbol{\theta} = \theta)| \rightarrow 0, \text{for } J \rightarrow \infty;$$

also see McDonald (1982) and Holland and Rosenbaum (1986). For finite J, the analog to the large sample version of EI is that $\text{Cov}(X_j, X_k | \boldsymbol{\theta}) \approx 0$, which is mathematically idealized to *weak local independence* (weak LI) or, equivalently, *pairwise local independence;* that is,

$$\text{Cov}(X_j, X_k | \boldsymbol{\theta} = \theta) = 0, \text{for all } \theta, \text{and for all } 1 \leq j < k \leq J \tag{1.2}$$

(Stout et al., 1996; Zhang & Stout, 1999a). Note that weak LI (Equation 1.2) is implied by LI (Equation 1.1), but not the other way around. In practice, weak LI may be used to investigate LI (Stout, 1990).

Weak monotonicity means that the average of J IRFs is an increasing function of $\boldsymbol{\theta}$, but leaves the individual IRFs unrestricted within the confines of this

condition on the mean; that is,

$$J^{-1}\sum_{j=1}^{J} P_j(\boldsymbol{\theta}_a) \leq J^{-1}\sum_{j=1}^{J} P_j(\boldsymbol{\theta}_b), \text{ whenever } \boldsymbol{\theta}_a < \boldsymbol{\theta}_b, \text{ coordinatewise.}$$

Thus, the strictly unidimensional model has a stronger independence assumption and a stronger monotonicity assumption than the essentially unidimensional model.

Discussion of the models. Although both have different points of departure, the essentially and strictly unidimensional IRT models both imply weak LI. For analyzing empirical data both types of models may use this property. For example, in the strictly unidimensional Rasch model the LI assumption is investigated for empirical test data using statistical tests based on weak local independence (Molenaar, 1983; also, see Glas & Verhelst, 1995). The most pronounced difference between the strictly and essentially unidimensional NIRT model discussed here is the investigation of the dimensionality of the responses to a set of items. Item selection based on strictly unidimensional models aims at finding one or more homogeneous (i.e., measuring one θ each) clusters, using observable consequences of the model of monotone homogeneity, in particular, of assumption M. Item selection based on essentially unidimensional models aims at finding clusters of items sensitive to one dominant trait each, using observable consequences of weak LI. These differences will be explained in the next sections in more detail.

1.2.2 Methods for Investigating Dimensionality

MSP

Let a set of items consist of J dichotomous items and let a unidimensional cluster of items consist of L items ($j = 1, \ldots, L; L \leq J$). The computer program Mokken Scale analysis for Polytomous items (MSP5 for Windows, MSP for short; Molenaar & Sijtsma, 2000) uses scalability coefficient H (Loevinger, 1948; Mokken, 1971) as the criterion for selecting items that yield a unidimensional cluster. For items j and k, the H coefficient is defined as the ratio of the covariance between items j and k, and their maximum covariance given the marginal distributions of the items; that is,

$$H_{jk} = \frac{\text{Cov}(X_j, X_k)}{\text{Cov}(X_j, X_k)_{\max}}.$$

Thus, H_{jk} is the normed covariance of an item pair. The scalability coefficient of a single item j with respect to the other $L - 1$ items selected into a cluster is

defined as

$$H_j = \frac{\sum_{k \neq j} \text{Cov}(X_j, X_k)}{\sum_{k \neq j} \text{Cov}(X_j, X_k)_{\max}}.$$

The item scalability coefficient H_j can be interpreted as an index for the slope of the IRF of item j. For example, under the 2-parameter logistic model (2-PLM; e.g., Birnbaum, 1968), fixing the distribution of θ and also the 2-PLM location parameters of the IRFs, the H_js are an increasing function of the slope parameters (Mokken, Lewis, & Sijtsma, 1986).

Finally, for a set of L items the scalability coefficient H is a weighted average of the item H_js, with positive weights depending on the marginals. Let π_j be the proportion correct on item j, and write $\text{Cov}(X_j, X_k)_{max} = \pi_{jk}^{(0)}$. Note that $\pi_{jk}^{(0)} = \pi_j(1 - \pi_k)$ if $\pi_j \leq \pi_k$; and $\pi_{jk}^{(0)} = \pi_k(1 - \pi_j)$ if $\pi_k < \pi_j$. Mokken (1971, p. 152) writes coefficient H as

$$H = \frac{\sum_{j=1}^{L-1} \sum_{k=j+1}^{L} \pi_{jk}^{(0)} H_j}{\sum_{j=1}^{L-1} \sum_{k=j+1}^{L} \pi_{jk}^{(0)}}. \tag{1.3}$$

Because fixed π_js also imply fixed $\pi_{jk}^{(0)}$s, an increase of the H_js causes an increase of H. Under UD, LI and M, it can be shown that $0 \leq H \leq 1$ (Mokken, 1971; p. 150). Given UD, LI, and M, the value of $H = 0$ means that the IRFs of at least $(L - 1)$ items are constant functions of θ, and $H = 1$ means that there are no Guttman errors (given that $\pi_j \leq \pi_k$, a Guttman error is defined as $X_j = 1$ and $X_k = 0$); see Mokken (1971, p. 150) for further elaboration. Mokken (1971, p. 184) defined a scale as follows:

DEFINITION: A cluster of items is a *Mokken* scale if,

$$Cov(X_j, X_k) > 0, \text{for all item pairs } (j, k; j \neq k); \text{and} \tag{1.4}$$

$$H_j \geq c > 0, \text{for all items } j, \tag{1.5}$$

where c is a positive lower bound of H_j, which is user-specified. The higher c, the more restrictive item selection is with respect to the discrimination of the items. A high c means good item discrimination and accurate person ordering using X_+ (also, see Sijtsma & Molenaar, 2002, p. 68).

MSP uses a sequential bottom-up item clustering procedure to partition a multidimensional set of items into clusters of items that each constitute a Mokken

scale. The default start set is the item pair in the pool with the highest significant positive H_{jk} (for other possibilities, see Molenaar & Sijtsma, 2000, chap. 5). The second step is the selection of an item from the remaining items, that satisfies Equations 1.4 and 1.5 with respect to the previously selected items, and maximizes the common H of the already selected items and the newly selected item. In the next steps, items are added to the already selected cluster using the same procedure. A scale has been completed when no more items remain that satisfy Equations 1.4 and 1.5. If items remain unselected, subsequent clusters of items may be selected as described for the first cluster. The procedure stops when no more items remain that satisfy Equations 1.4 and 1.5. For more details about the item selection procedure, see Hemker et al. (1995) and Molenaar and Sijtsma (2000).

Additional remarks. First, by selecting Mokken scales using scaling condition $H_j \geq c$ the dimensionality of the data is implicitly investigated as well (see Hemker et al., 1995). Consider the following idealized situation. Assume that some items are driven by θ_1 and other items by θ_2, and that these traits are correlated. Notice that, for the entire set of items an IRF is the regression of X_j on a composite of these two θs, and that H_j expresses the strength of this relationship. Finally, assume that the relationship of the items driven by θ_1 with θ_1 is stronger than that of the items driven by θ_2 with θ_2. The rest score, $R_{(-j)} = X_+ - X_j$, estimates the latent trait composite, and the regression of item j on $R_{(-j)}$ is given by $P[X_j = 1|R_{(-j)}]$. Based on these assumptions, in general, the regression of items driven by θ_1 on $R_{(-j)}$ is steeper (higher H_j) than that of the items driven by θ_2 (lower H_j).

Suppose that the item pair selected first is driven by θ_1, then a conveniently chosen c value selects the other items sensitive to θ_1 into the first cluster because their H_js with respect to the already selected items are greater than those of items sensitive to θ_2. If these latter items have $H_j s < c$, they remain unselected and the first item cluster is completed. Because the remaining items are driven by θ_2, rest score $R_{(-j)}$ based on these items estimates θ_2 and the regression, $P[X_j = 1|R_{(-j)}]$, is steeper resulting in higher H_js. If these H_js exceed lower bound c, then a second cluster consisting of items sensitive to θ_2 is selected.

The choice of lowerbound c affects the cluster composition. A low c value may result in clusters that are highly heterogenous with respect to latent trait composition. A high c value yields a cluster with high $H_j s$, but as a consequence many items sensitive to the same latent trait may be rejected. In general, when determining an appropriate value of c a researcher should find a balance between the number of items in a scale and the strength of the scale.

Second, because MSP uses a sequential item selection procedure, comparable to forward stepwise regression in SPSS (1998), not all combinations of items are considered. Therefore, the final item clusters may not have the maximum possible H coefficient for each cluster given all possible partitions of the total set. MSP offers a possibility to refine the search procedure; see Mokken (1971, pp. 198-199) and Sijtsma and Molenaar (2002, p. 72) for more details.

DETECT

Let composite θ_α be a linear combination of the separate θs from latent trait vector $\boldsymbol{\theta}$ (which may contain several dominant traits and several nuisance traits simultaneously). Composite θ_α can be understood as the latent direction that is best measured by the test (see, Zhang & Stout, 1999a, for a rigorous definition of the direction of best measurement of a test). Given unidimensionality, following Equation 1.2, the expected conditional covariance of an item pair equals 0. If θ_α is built up from multiple traits differentially measured by different items, the expected conditional covariance is positive when items j and k are driven by the same latent trait or traits that correlate highly, and negative when items j and k are driven by traits that correlate weakly or zero. The computer program DETECT uses the sign behavior of the conditional covariances to find clusters of dimensionally homogeneous items.

More specifically, DETECT (Kim, 1994; Zhang, 1996; Zhang & Stout, 1999b) partitions, as much as possible, the set of items into an a priori specified maximum number of clusters in such a way that the expected conditional covariances between items from the same cluster are positive and the expected conditional covariances between items from different clusters are negative. Consider an arbitrary partitioning $\mathcal{P}$ of the item pool. Let $\delta_{jk}(\mathcal{P}) = 1$ if items j and k are in the same cluster of $\mathcal{P}$; and $\delta_{jk}(\mathcal{P}) = -1$ otherwise (Zhang & Stout, 1999b). Then, the theoretical DETECT index is defined as

$$D_\alpha(\mathcal{P}) = \frac{2}{J(J-1)} \sum_{1 \le j < k \le J} \delta_{jk}(\mathcal{P}) E[\mathrm{Cov}\,(X_j, X_k | \theta_\alpha)]. \tag{1.6}$$

DETECT tries to find the partition that maximizes $D_\alpha(\mathcal{P})$. This partition is denoted as $\mathcal{P}^*$ and is taken as the final cluster solution. Thus, DETECT attempts to find dimensionally homogeneous clusters of items, each of which may be interpreted to assess another latent trait and, this way, DETECT finds the number of dominant latent variables within a data matrix. Because the number of possible partitions increases very fast with the number of items, DETECT uses

a genetic algorithm to search for the optimal partition. The criterion that is used to evaluate each partitioning is the DETECT index, $D_\alpha(\mathcal{P})$.

A geometrical representation (e.g., Ackerman, 1996; Stout et al., 1996), depicted in Figure 1.1, helps to visualize item response data driven by two θs. The vectors' length depends on the item discrimination, and the vectors' angles reflect the correlation between variables. Items j, k, l, m and n are differentially sensitive to both θs and item n exactly measures composite θ_α. In yielding a particular θ_α value, it is assumed that high values on one latent trait can compensate for low values on another. For any value of θ_α, we may project a line that has a 90° angle with vector θ_α. This projected line then indicates for which combinations of values for θ_1 and θ_2 that particular value of θ_α is found. Because of this compensation, for a fixed value of θ_α, the probability of correctly answering two items driven by one latent trait (e.g., items j and k, driven by θ_1) may be higher than expected under LI. That is, subjects with a particular θ_α value who answer item j positively are likely to answer item k also positively. The reverse may hold when items are driven by different traits (e.g., items k and l). Thus, the expected conditional covariance of an item pair is positive when the same dominant trait may have been measured, and negative when different traits have been measured.

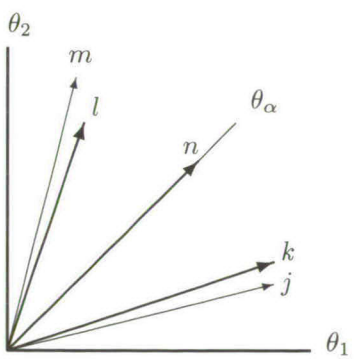


Figure 1.1: *Geometrical Representation for Two Traits and Five Items*

Let rest score $R_{(-j,-k)} = X_+ - X_j - X_k$ be the total score ignoring the two studied items j and k. The sample DETECT statistic uses the following estimate of the expected conditional covariances,

$$E\left[\widehat{\mathrm{Cov}}(X_j, X_k|\theta_\alpha)\right] = \frac{E\left\{\widehat{\mathrm{Cov}}[X_j, X_k|R_{(-j,-k)}]\right\} + E\left[\widehat{\mathrm{Cov}}(X_j, X_k|X_+)\right]}{2}. \quad (1.7)$$

This average of the expected covariances was used because $E[\widehat{\text{Cov}}(X_j, X_k|X_+)]$ tends to be negatively biased and $E\{\widehat{\text{Cov}}[X_j, X_k|R_{(-j,-k)}]\}$ positively biased (Junker, 1993; Zhang & Stout, 1999a). The average of the two expected conditional covariances was expected to be less biased (Zhang & Stout, 1999a).

Additional remarks. First, DETECT is relatively new and much theoretical research remains to be done. For example, the distribution of theoretical $D_\alpha(\mathcal{P})$ under interesting hypotheses is still unknown. In addition, in spite of Equation 1.7 the DETECT index still is slightly biased (e.g., Zhang, Yu, & Nandakumar, 2003 investigate bias for various DETECT indices).

Second, Zhang and Stout (1999b) showed that DETECT finds the correct partitioning if items are mainly sensitive to one trait and only marginally to other traits. This is know as approximate simple structure (see Zhang & Stout, 1999b for a rigorous definition). When data deviate from approximate simple structure, the correct dimensionality may not be found (Zhang & Stout, 1999b).

Third, the DETECT index expresses the magnitude of the departure from unidimensionality within one or more clusters of the partition but is not an index of the number of traits within the item response data. Thus, there may be a high number of dimensions and yet $D_\alpha(\mathcal{P})$ is small, or there may be few dimensions and yet $D_\alpha(\mathcal{P})$ is large.

HCA/CCPROX

The software package HCA/CCPROX (Roussos et al., 1998) uses agglomerative hierarchical cluster analysis (HCA) for finding clusters of items. The program provides the opportunity to choose between different statistics, including conditional covariances, for assessing the relationship between variables. The user can also choose between different agglomerative HCA methods. Only the combination of statistic and method that according to Roussos et al. (1998) was most successful in dimensionality assessment is presented here.

The program starts with each of the J items as a separate cluster. Then, at the second level of hierarchy, the two items having the smallest expected conditional covariance, $E\{\text{Cov}[X_j, X_k|R_{(-j,-k)}]\}$, are joined. For the subsequent steps we introduce some additional notation. In general, at one particular step in the clustering process, let A_v and A_w denote two clusters of items, containing J_v and J_w items, respectively. Let $R_{(-A_v,-A_w)}$ denote the rest score, containing all responses to items that are not in A_v and A_w. Then, we may define the expected conditional covariance, $E\{\text{Cov}[X_i, X_j|R_{(-A_v,-A_w)}]\}$. In each of the subsequent levels of hierarchy that pair of clusters is joined that is closest of all pairs according

to the proximity measure,

$$\text{Prox}(A_v, A_w) = (J_v J_w)^{-1} \sum_{i \epsilon A_v} \sum_{j \epsilon A_w} |E[\text{Cov}(X_i, X_j | R_{(-A_v, -A_w)})]|.$$

The process of joining clusters is repeated until all J items are collected into one large cluster.

Additional remarks. First, HCA/CCPROX does not provide a formal criterion, such as the lower bound c of coefficient H in MSP or the maximum DETECT index $D_\alpha(\mathcal{P}^*)$, that helps the researcher to decide which one of the $J-1$ possible cluster outcomes reflects the true dimensionality best. Consequently, the researcher must choose the solution that most likely represents the dimensionality of the item response data. Due to the lack of a formal criterion, the researcher should rely on a priori theoretical expectations about the true dimensionality structure of the data. For example, when it is expected that a verbal test measures vocabulary, grammar, and spelling, and each item is assumed to predominantly measure one trait, then the three-cluster solution from HCA/CCPROX is appropriate here.

Second, according to Roussos et al. (1998) the positively biased $E\{\widehat{\text{Cov}}[X_i, X_j | R_{(-A_v, -A_w)}]\}$ will not affect the cluster analysis much, because two items sensitive to different traits have an expected conditional covariance that is larger than that of two items that are sensitive to the same latent trait. HCA/CCPROX should therefore be able to correctly partition the items according to their dimensionality.

DIMTEST

DIMTEST is a statistical test procedure that evaluates the unidimensionality of data from a user-specified item set (Nandakumar & Stout, 1993; Stout, 1987; Stout et al., 2001). The procedure of DIMTEST is the following. First, the item pool is split into three subtests, of which two are assessment subtests (denoted AT1 and AT2) and one is a partitioning subtest (denoted PT). One may use factor analysis or, for example, MSP or DETECT to have a sensible basis for AT1, AT2 and PT. DIMTEST provides linear factor analysis on the tetrachoric correlation matrix to determine which M items out of the total set of N items (the number M is user-specified; for rules of thumb, see Nandakumar & Stout, 1993) are selected in AT1. These M items that constitude AT1 are hypothesized to be sensitive to the same trait. AT2 consists of M items sensitive to another trait than that measured by AT1, but with a similar observed frequency distribution of proportions correct on the items. Subtest PT is formed using the $J - 2M$ remaining items.

Using the sum scores on the PT subtest, the group of examinees is partitioned into subgroups of at least 20 (as recommended by Stout, 1987) of approximately equal ability. AT2 is designed to reduce 'examinee variability bias' (i.e., θ still has a positive variance given a fixed PT score) and 'item difficulty bias' (i.e., θ variance is inflated even more when items in the AT1 test and the PT test vary in difficulty). For short tests both kinds of bias may inflate the DIMTEST statistic enough to incorrectly reject the null hypothesis of unidimensionality.

Let X_j^{AT1} and X_k^{AT1} be the scores on two items from AT1; and let Y_{PT} be a total score comparable with X_+ based on all items in PT. The DIMTEST sample statistic is based upon,

$$\mathrm{Cov}\left(X_j^{AT1}, X_k^{AT1} | Y_{PT} = y\right). \tag{1.8}$$

Under unidimensionality and for large J, this covariance must be close to zero for any item pair from AT1 and any Y_{PT} score. Under regularity conditions, the original DIMTEST-statistic T (Stout, 1987), and the more powerful T' (Nandakumar & Stout, 1993) are distributed asymptotically (both in N and J) standard normally when unidimensionality holds. Given a significance level α and the upper $100(1-\alpha)$ percentile of a standard normal distribution, Z_α, unidimensionality is rejected when $T > Z_\alpha$ or $T' > Z_\alpha$.

Additional remarks. First, DIMTEST tests the specific hypothesis that unidimensionality holds in a particular data set. For that reason DIMTEST, unlike MSP, DETECT and HCA/CCPROX, cannot directly be used to partition items in different clusters. Second, DIMTEST exhibits some positive bias because of the use of test scores as conditioning variable even after correcting for two types of bias using AT2. Third, Stout et al. (2001) proposed a new DIMTEST procedure which uses only one subtest AT. The aim of the new DIMTEST procedure is to further reduce bias and increase power of T'. The properties of the new procedure are still subject to investigation. Therefore, we did not use it in this study.

1.3 Simulation Study

A simulation study was done to compare the effectiveness of MSP, DETECT, and HCA/CCPROX for selecting items into clusters that represent the true dimensionality of the data. Also, it was investigated whether the DETECT statistic, $D_\alpha(\mathcal{P})$, and the DIMTEST statistic, T', indicate whether the true model is essentially unidimensional or multidimensional. The simulation study involved six factors: (1) the IRT model used for simulating the data (two models), (2) the number of latent traits (two numbers), (3) the correlation between the latent traits

(six correlations), (4) the number of items per trait (for each number of latent traits, four combinations of numbers of items), (5) the item discrimination per trait (three combinations), and (6) the item selection method (four methods). For each cell of the $2 \times 2 \times 6 \times 4 \times 3 \times 4$ design, 2,000 simulees were generated from a multivariate standard normal density. Data were simulated assuming simple structure (Stout, et al., 1996), meaning that items loaded only on one trait, but traits were allowed to correlate. Part of the design was replicated five times to investigate the stability of the results. For a few cells of the design, a smaller sample size ($N = 200$) was investigated.

IRT model. To simulate multidimensional item response data, the multidimensional extensions of the 2-PLM and the five-parameter acceleration model (5-PAM; see also Sijtsma & Van der Ark, 2001; Samejima, 1995; 2000) were used. Several researchers (e.g., Hemker et al., 1995; Reckase & McKinley, 1991; Roussos et al., 1998) used the 2-PLM for simulating data, but we also simulated data using the more general 5-PAM to allow IRFs to take on a more flexible shape. Let $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_Q)$ be the vector of Q latent traits (no nuisance traits); and let θ_{iq} be the value of person i on trait q. The 5-PAM has five item parameters: let α_{jq} be the discrimination parameter of item j on trait q ($q = 1, \ldots, Q$); δ_{jq} the location parameter of item j on trait q; γ_j^{up} and γ_j^{lo} the upper and lower asymptotes of the IRF, respectively; and ξ_j the acceleration parameter. Then, for a multidimensional extension of the 5-PAM, to be denoted M5-PAM, the probability of answering item j correctly, given the latent trait vector $\boldsymbol{\theta}$, is

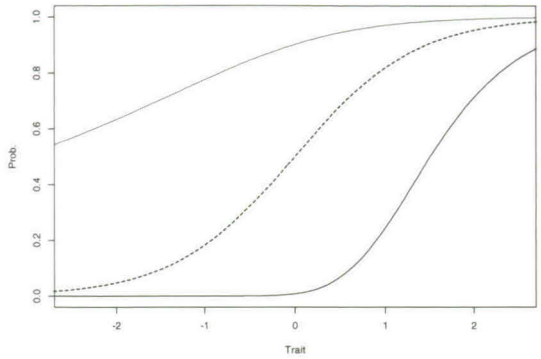
$$P(X_j = 1|\boldsymbol{\theta}) = \gamma_j^{lo} + (\gamma_j^{up} - \gamma_j^{lo}) \left\{ \frac{\exp\left[\sum_{q=1}^{Q} 1.7\alpha_{jq}(\theta_{iq} - \delta_{jq})\right]}{1 + \exp\left[\sum_{q=1}^{Q} 1.7\alpha_{jq}(\theta_{iq} - \delta_{jq})\right]} \right\}^{\xi_j}. \quad (1.9)$$

Parameter γ_j^{lo} and parameter γ_j^{up} allow the lower asymptote to be larger than 0 and the upper asymptote to be smaller than 1, respectively. Parameter ξ_j allows the IRF to be asymmetric (see also Samejima, 1995; 2000). The multidimensional 2-PLM (M2-PLM) (also, see Reckase, 1997) is a special case of the M5-PAM for $\gamma_j^{lo} = 0$, $\gamma_j^{up} = 1$ and $\xi_j = 1$. For illustration of the effect of ξ in the 5-PAM items, see Figure 1.2.

Number of traits. The numbers of latent traits used here were two and four.

Correlation between traits. The six product-moment correlations (ρ) between the latent traits were 0.0, 0.2, 0.4, 0.6, 0.8, and 1.0. The correlation of 0.0 represents independent latent traits, and the correlation of 1.0 represents unidimensionality.

Figure 1.2: Illustration of the effect of ξ on the shape of 5-PAM IRFs: $\xi_j = 0.15$ (top), $\xi_j = 1$ (middle), and $\xi_j = 7$ (bottom) and other parameter values are $\alpha_j = 1.5$, $\delta_j = 0$, $\gamma_j^{up} = 1$ and $\gamma_j^{lo} = 0$.



Number of items per trait. For $Q = 2$ and $Q = 4$, four different combinations of the number of items per trait were chosen. Each trait was measured by either a small or a large number of items. For $Q = 2$, the four different combinations of test lengths within the item pool were: short - short; short - long; long - short; and long - long. We used notation $[2{:}v; w]$ to indicate that two latent traits were generated with v items sensitive to θ_1 and w items sensitive to θ_2. Likewise, $[4{:}v; w; y; z]$ is the four-dimensional extension of this notation. For $Q = 2$, the four combinations were [2:7;7], [2:7;21], [2:21;7], and [2:21;21]; and for $Q = 4$, the four combinations were [4:7;7;7;7], [4:7;7;21;21], [4:21;21;7;7], and [4:21;21;21;21]. Each of these eight simulated combinations of number of items per trait is referred to as the 'true dimensional structure' or 'simulated dimensional structure'. It may be noted that by varying the number of items per trait across design cells, the total number of items in the item pool across design cells also varies.

Discrimination per trait. All items measuring the same latent trait either had low discrimination or high discrimination. If items all had low discrimination, the discrimination parameters were sampled from a distribution, to be discussed shortly, in such a way that discrimination varied but was low for all items. The same procedure was followed for items having high discrimination. Once the parameters had been sampled, they were fixed across the design cells for which the discrimination level was held constant. Information referring to high discrimination items is printed in boldface. For example, for $Q = 2$ and 7 items per subset, three combinations of discrimination were used: [2:7;7], [2:7;**7**], and [2:**7**;**7**]; and for $Q = 4$, the combinations were [4:7;7;7;7], [4:7;7;**7**;**7**], and [4:**7**;**7**;**7**;**7**].

Item discrimination was operationalized as the maximum slope of the IRF. In the special case of the M2-PLM, this maximum equals the discrimination parameter α_{jq}, but in the M5-PAM the slope also depends on parameters γ_j^{lo}, γ_j^{up}, and ξ_j. Thus, in the M5-PAM, the maximum slope (α_{jq}^*) was calculated using the first partial derivative of Equation 1.9. This resulted in

$$\begin{aligned} \alpha_{jq}^* &= \frac{4}{1.7}\left[\max\left(\frac{\partial P_j(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)\right] \\ &= \frac{4}{1.7}\left[\alpha_{jq}\xi_j(\gamma_j^{up}-\gamma_j^{lo})\left(\frac{\xi_j}{1+\xi_j}\right)\left(1-\frac{\xi_j}{1+\xi_j}\right)\right]. \end{aligned} \tag{1.10}$$

From Equation 1.10 it follows that,

$$\alpha_{jq} = \frac{\alpha_{jq}^*}{\frac{4}{1.7}\left[\xi_j(\gamma_j^{up}-\gamma_j^{lo})(\frac{\xi_j}{1+\xi_j})(1-\frac{\xi_j}{1+\xi_j})\right]}. \tag{1.11}$$

Thus, α_{jq} can be calculated when $\gamma_j^{lo}, \gamma_j^{up}, \xi_j$, and α_{jq}^* are known. Constant 4/1.7 is included in Equation 1.11 so that in the M2-PLM $\alpha_{jq}^* = 1.7 \times \alpha_{jq}$. Thus, α_{jq} depends on $\gamma_j^{lo}, \gamma_j^{up}, \xi_j$, and α_{jq}^*.

Parameters $\gamma_j^{lo}, \gamma_j^{up}$, and ξ_j influence the location of θ where α_{jq}^* reaches its maximum. If δ_{jq}^* is the location where the M5-PAM item discriminates best, then the corresponding location parameter equals

$$\delta_{jq} = \delta_{jq}^* - \frac{\ln(\xi_{jq})}{\alpha_{jq}^*}. \tag{1.12}$$

The parameters were generated to resemble parameter estimates found in analysis of real test data. Under the M2-PLM, for items with low discrimination, α_{jq} is the exponentiation of a number randomly drawn from a normal distribution with mean 0.75 and variance 0.1, truncated at 0.5 and 1.25. For items with high discrimination, α_{jq} is the exponentiation from a number randomly drawn from a normal distribution with mean 1.75 and variance 0.1, truncated at 1.5 and 2.25. The difficulty parameters were chosen equidistant between -2.0 and 2.0.

Under the M5-PAM, γ_j^{lo} was chosen from the interval between 0.0 and 0.2, γ_j^{up} was chosen between 0.8 and 1.0, and ξ_j between 0.5 and 7, such that the slope (α_{jq}^*) and the location (δ_{jq}^*) under the M2-PLM and the M5-PAM were mathematically equal. However, the different shapes of the curves may prevent a direct and easy comparison of the results generated under the two models.

Item selection method. For the three item selection procedures, MSP, DETECT, and HCA/CCPROX, and for DIMTEST, the default settings were used as much as possible. Also, the recommendations made by the authors in various papers were used.

For MSP, we used the default lower bound value of $c = 0.30$ (Molenaar & Sijtsma, 2000). In addition, following recommendations by Hemker et al. (1995), for a part of the design we investigated the influence of different c-values (0.10, 0.20, 0.30, 0.40, and 0.50) on the retrieval of the true dimensionality structure.

For DETECT, DIMTEST, and HCA/CCPROX, stable conditional covariance estimates were obtained using the item-score vectors of at least 20 simulees per estimated θ_α (Stout, 1987) unless this led to the rejection of more than 15 percent of the item score vectors. Then, the minimum group size was lowered to 10.

For DIMTEST, factor analysis of 500 item score vectors determined which items were used in AT1. The remaining 1500 item score vectors were used to calculate the DIMTEST statistic. As recommended by Nandakumar and Stout (1993), the number of items M included in AT1 was determined by the rules that $4 < M \leq J/4$ and the absolute value of the loadings $\geq .15$. In the 14-item tests we used $M = 3$.

1.4 Results

1.4.1 Comparison of the Item Selection Methods

In the notation [4: $v, w; y; z$], the first number (here, 4) reflects the number of clusters found either by MSP, DETECT or HCA/CCPROX; v reflects the number of items selected into the first cluster; w reflects the number of items selected into the second cluster; and so on. A semicolon separates two clusters that are sensitive to different latent traits. A comma separates two clusters that are sensitive to the same latent trait. A classification error is defined as two items in the same cluster are sensitive to different latent traits. Such errors are denoted by a slash as in [2:7/7], meaning that at least one of the two clusters contains items that are sensitive to different θs.

We distinguish five types of results. *Type A* means all J items were selected into the true dimensional structure. *Type B* indicates that the correct number of clusters and no classification errors were found, but not all J items were selected. *Type C* reflects that the true dimensionality was found to a high degree, but the number of clusters was larger than the Q latent traits in the sense that two or more clusters were driven by the same trait. Thus, types A, B, and C do not have classification errors. *Type D* reflects that the true dimensional structure was not found; that is, items driven by different latent traits were selected into one subset. *Type E* represents the result where all items were selected into one subset. Types D and E have classification errors. For $\rho = 1.0$, Type E is the correct outcome

and for $\rho = 0.0$ Type A is the correct outcome.

Two-dimensional data sets based on M2-PLM

Correlation between traits. Table 1.1 shows that as correlations between traits (ρ) increased, the simulated dimensional structure was found less often by each of the item selection procedures.

Interaction of Correlation between traits $\times$ *Method.* The effect of increasing ρ on item selection was more apparent in MSP than in DETECT and HCA/CCPROX. For example, MSP found the simulated structure in [2:**7**;**7**] for $\rho = 0.0$ and $\rho = 0.2$, and as ρ increased MSP tended to select more items sensitive to different traits into the same cluster until for $\rho = 1$ a Type E result was found. These classification errors are made when the inter-item correlations are such that lowerbound c is not restrictive enough to split items sensitive to different traits into different clusters. DETECT and HCA/CCPROX found the simulated structure approximately until $\rho = 0.8$. Table 1.1 shows that for highly correlating traits, DETECT continued to form multiple clusters, even when items correlated $\rho = 1.0$. Due to sampling fluctuations and a weakly biased $D_\alpha(\mathcal{P})$-statistic, the observable conditional covariances were nonzero, even when the data were unidimensional. For these reasons, $D_\alpha(\mathcal{P})$ can be highest for a partitioning having two or more clusters.

Discrimination. With increasing α^*_{jq}, the simulated dimensional structure was found more often for each of the item selection methods; see Table 1.1.

Interaction of Discrimination $\times$ *Method.* MSP was more sensitive to item discrimination than DETECT and HCA/CCPROX. Variation in mean α^* between latent traits within one data matrix was also simulated. Latent traits that were represented by clusters of weakly discriminating items were not well recovered by any of the three item selection methods, but latent traits that were represented by means of highly discriminating items were well recovered.

Number of items per trait. Traits represented by seven items were, in general, equally well recovered as traits represented by 21 items.

Interaction of Number of items per trait $\times$ *Method.* For clusters containing 21 items having low item discrimination, MSP sometimes misclassified a single item out of the total set. Another result was that MSP selected the lowly discriminating items into an extra cluster (i.e., Type C). Such results were not found for latent traits assessed by 7 items. DETECT produced more Type C results in the unequal number of items condition compared to the equal conditions. HCA/CCPROX produced approximately the same results irrespective of the number of items per trait.

Table 1.1: *Item Selection Results Using the M2-PLM and Two Latent Traits*

Test	MSP					
Composition ρ :	0.0	0.2	0.4	0.6	0.8	1.0
[2 : 7; 7]	[3:2,5;6]	[3:2,5;7]	[2:7;6]	[3:2/3/7]	[4:2/2/2/8]	[2:10/2]
[2 : 7; 21]	[2:6;19]	[4:2,5;2,19]	[5:2,5;2,17]	[4:2/2/3/20]	[4:2/2/2/21]	[3:2/2/24]
[2 : 21; 7]	[3:19,2;7]	[3:19,2;5]	[2:20/5]	[3:20/4/2]	[3:22/2/2]	[2:25/2]
[2 : 21; 21]	[4:2,18;2,19]	[3:2,18;19]	[4:2,18; 2,19]	[4:2/2/9/27]	[5:2/2/2/2/31]	[2:2/39]
[2 : 7; **7**]	[3:2,5;7]	[2:7;6]	[2:6;7]	[1:13]	[1:14]	[1:14]
[2 : 7; **21**]	[2:6;21]	[2:7;21]	[2:5;21]	[2:2/25]	[1:27]	[1:28]
[2 : 21; **7**]	[3:2,18;7]	[3:2,18;7]	[4:2,2,17;7]	[2:2/26]	[1:27]	[1:27]
[2 : 21; **21**]	[3:2,19;21]	[3:2,18;21]	[3:2/17/23]	[3:2/2/37]	[2:2/40]	[1:42]
[2 : **7; 7**]	[2:7;7]	[2:7;7]	[1:14]	[1:14]	[1:14]	[1:14]
[2 : **7; 21**]	[2:7;21]	[2:7;21]	[1:28]	[1:28]	[1:28]	[1:28]
[2 : **21; 7**]	[2:21;7]	[2:21;7]	[1:28]	[1:28]	[1:28]	[1:28]
[2 : **21; 21**]	[2:21;21]	[2:21;21]	[1:42]	[1:42]	[1:42]	[1:42]

Note: Boldface indicates highly discriminating items. Bracket notation: a *semicolon* separates dimensionally different clusters; a *comma* separates dimensionally similar clusters; and a *slash* separates mixed clusters.
Table continues on the next page.

Table 1.1: (continued)

Test Composition ρ:	DETECT 0.0	0.2	0.4	0.6	0.8	1.0
[2 : 7; 7]	[2:7;7]	[2:7;7]	[2:7;7]	[2:7;7]	[3:3/5/6]	[5:2/2/2/2/6]
[2 : 7; 21]	[2:7;21]	[2:7;21]	[3:7;1,20]	[2:7;21]	[4:7;1,6,14]	[4:4/5/6/13]
[2 : 21; 7]	[2:21;7]	[2:21;7]	[2:21;7]	[3:2,19;7]	[4:2,2,19;7]	[4:3/10/10/5]
[2 : 21; 21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[4:1/12/12/17]
[2 : 7; **7**]	[2:7;7]	[2:7;7]	[2:7;7]	[2:7;7]	[3:1,6;7]	[3:2/3/9]
[2 : 7; **21**]	[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[4:2,2,3;21]
[2 : 21; **7**]	[2:21;7]	[2:21;7]	[2:21;7]	[4:1,2,18;7]	[4:4,8,9;7]	[4:3/3/4/18]
[2 : 21; **21**]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[3:5/8/29]
[2 : **7; 7**]	[2:7;7]	[2:7;7]	[2:7;7]	[2:7;7]	[2:7;7]	[1:14]
[2 : **7; 21**]	[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[3:3/11/14]
[2 : **21; 7**]	[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:10;18]
[2 : **21; 21**]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[3:18/16/8]

Note: Boldface indicates highly discriminating items. Bracket notation: a *semicolon* separates dimensionally different clusters; a *comma* separates dimensionally similar clusters; and a *slash* separates mixed clusters.

Table continues on the next page.

Table 1.1: (continued)

Test		HCA/CCPROX					
Composition	ρ :	0.0	0.2	0.4	0.6	0.8	1.0
[2 : 7; 7]		[2:7;7]	[2:7;7]	[2:7;7]	[2:7;7]	[2:1/13]	[2:2/12]
[2 : 7; 21]		[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:3/25]
[2 : 21; 7]		[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:4/24]
[2 : 21; 21]		[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:2/40]
[2 : 7; **7**]		[2:7,7]	[2:7;7]	[2:7;7]	[2:7;7]	[2:7;7]	[2:2/12]
[2 : 7; **21**]		[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:6/22]
[2 : 21; **7**]		[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:4/24]
[2 : 21; **21**]		[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:9/32]
[2 : **7; 7**]		[2:7;7]	[2:7;7]	[2:7;7]	[2:7;7]	[2:2/12]	[2:2/12]
[2 : **7; 21**]		[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:7;21]	[2:10/18]
[2 : **21; 7**]		[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:21;7]	[2:3/25]
[2 : **21; 21**]		[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:21;21]	[2:5/37]

Note: Boldface indicates highly discriminating items. Bracket notation: a *semicolon* separates dimensionally different clusters; a *comma* separates dimensionally similar clusters; and a *slash* separates mixed clusters.

Method. In general, the simulated structure was found more often by DETECT and HCA/CCPROX than by MSP. HCA/CCPROX results should be interpreted with care because we only presented the outcomes when the number of clusters equalled the number of simulated traits (Q). In practical data analysis, however, the researcher has to decide which cluster solution is best, possibly relying on previous knowledge about the trait structure of the data. Thus, the results of HCA/CCPROX presented here and elsewhere in the results section may be more favorable than in practical data analysis. For $\rho = 1.0$, the HCA/CCPROX partitioning only reflects random fluctuation.

Replications based on M2-PLM

For [2:7;7], [2:7;21], and [2:**7**;**7**]; for $\rho = 0.0, 0.4$, and 0.8; and for MSP, DETECT, and HCA/CCPROX, five data matrices were randomly and independently sampled (results are not presented in a table). True dimensionality was found consistently across replications, in particular for highly discriminating items and low correlations between traits. DETECT and HCA/CCPROX yielded more consistent results than MSP. This may be due to the scaling condition $H_j \geq c$ in MSP. For some items this condition may be satisfied in some samples but not in others resulting in different cluster-solutions between samples. DETECT and HCA/CCPROX do not have such a scaling condition and the effect of sample fluctuations on the cluster-solution may therefore be smaller. In other design cells also included in the replication investigation, MSP and DETECT often found an extra cluster, and HCA/CCPROX misclassified several items.

Small sample size

The MSP and HCA/CCPROX results for $N = 200$ and $N = 2,000$ were approximately the same in the design cells for [2:7;7], [2:7;21], and [2:**7**;**7**]; and $\rho = 0.0, 0.4$, and 0.8. DETECT's results were somewhat worse for $N = 200$, probably due to inaccurate conditional covariance estimates in too small X_+ and $R_{(-j,-k)}$ score groups. MSP uses the H_j coefficient, which is based on the whole sample and, therefore, is more stable.

Four-Dimensional Simulation Using the M2-PLM

In general, the results for $Q = 2$ and $Q = 4$ (Table 1.2) were comparable. However, for $Q = 4$ more results of Type B and Type C were found (A, B, C, D, E notation is used to save space), because the greater number of items gave rise to more chance capitalization. A peculiar result for DETECT was that for [4:7;7;21;21] and

[4:21;21;7;7], as ρ increased, DETECT (but not HCA/CCPROX) selected the two clusters of seven equally discriminating items, sensitive to different latent traits with equal discrimination, into one cluster. The effect was more pronounced for higher discrimination. For HCA/CCPROX, only the correct (Type A) or incorrect (Type D) solutions were reported because of the use of the foreknowledge that $Q = 4$.

Two-Dimensional Simulation Using the M5-PAM

For data generation using the M5-PAM, only those factor levels were used that proved to be informative in the M2-PLM analysis: 2 traits (not 4); either low or high discrimination (maximum slope α^*) (no combination); 7 or 21 items per trait; and correlations between the traits that varied from 0.0 to 1.0. The design, therefore, had the order 2 (discrimination levels) $\times$ 2 (number of items per trait) $\times$ 6 (correlation between traits) $\times$ 4 (item selection method).

The general trend in the results (Table 1.3) was the same as with simulation using the M2-PLM. For any of the three methods, for a higher ρ and a lower α^* the dimensional structure was found less often (see Table 1.3). As before, these trends were more obvious for MSP than for DETECT and HCA/CCPROX. For the number of items per cluster, the effects were reversed: for 21-item clusters somewhat better results were obtained than for 7-item clusters. However, the differences were small and may be due to sample fluctuation. As for the M2-PLM, DETECT found the simulated dimensionality less often for unequal numbers of items.

Compared to the M2-PLM, in general all three methods performed a little worse. For MSP more Type B results were found, for DETECT more Type C results were found, and for HCA/CCPROX more Type D results were found (cf. Tables 1.1 and 1.3). These results may, in part, be due to the different overall shapes of the IRFs of the M5-PAM and the M2-PLM. Even when two IRFs from different models have equivalent maximum slopes (and equal locations), their slopes are not the same for all θs. In this study, this resulted in a somewhat lower overall discrimination for the M5-PAM items. This might explain that more minor deviations from the simulated dimensional structure were found when using the M5-PAM than when using the M2-PLM.

Manipulating Lower bound c in Mokken Scale Analysis

The effect of using different c-values (0.00, 0.20, 0.30, 0.40, and 0.50; Hemker et al., 1995) on MSP item selection was investigated for several design cells; that

Table 1.2: *Four-dimensional Item Selection Results Using the Multidimensional Two-Parameter Logistic Model (M2-PLM)*

Test		MSP						DETECT						HCA/CCPROX					
Composition	ρ:	.0	.2	.4	.6	.8	1	.0	.2	.4	.6	.8	1	.0	.2	.4	.6	.8	1
[4 : 7; 7; 7; 7]		B	C	B	D	D	D	A	A	A	A	A	D	A	A	A	A	D	D
[4 : 7; 7; 21; 21]		C	C	C	D	D	D	D	D	A	D	D	D	A	A	A	A	D	D
[4 : 21; 21; 7; 7]		C	C	C	D	D	D	A	D	D	A	A	D	A	A	A	A	A	D
[4 : 21; 21; 21; 21]		C	C	D	D	D	D	A	A	D	A	A	D	A	A	A	A	D	D
[4 : 7; 7; **7**; **7**]		C	C	D	D	D	E	A	A	A	A	D	D	A	A	A	A	D	D
[4 : 7; 7; **21**; **21**]		B	C	C	D	E	E	A	D	D	D	D	D	A	A	A	D	D	D
[4 : 21; 21; **7**; **7**]		C	C	D	D	D	D	A	D	D	A	A	D	A	A	A	A	D	D
[4 : 21; 21; **21**; **21**]		C	B	D	D	D	E	A	A	A	A	A	D	A	A	A	A	A	D
[4 : **7**; **7**; **7**; **7**]		A	A	D	E	E	E	A	A	A	A	A	D	A	A	A	A	A	D
[4 : **7**; **7**; **21**; **21**]		A	A	D	E	E	E	A	A	D	A	A	D	A	A	A	D	D	D
[4 : **21**; **21**; **7**; **7**]		A	A	E	E	E	E	A	D	D	D	D	D	A	A	A	A	D	D
[4 : **21**; **21**; **21**; **21**]		A	A	E	E	E	E	A	D	A	A	A	D	A	A	A	D	A	D

Note: Boldface indicates highly discriminating items; A='true dimensionality found'; B='not all items included';C='multiple clusters'; D='dimensionality not found'; and E='all items in one subset'.

Table 1.3: *Two-Dimensional Item Selection Results Using the Multidimensional Five-Parameter Acceleration Model (M5-PAM)*

Test		MSP						DETECT						HCA/CCPROX					
Composition	ρ:	.0	.2	.4	.6	.8	1	.0	.2	.4	.6	.8	1	.0	.2	.4	.6	.8	1
[2 : 7; 7]		B	B	B	E	E	E	A	A	A	C	C	D	A	A	A	A	D	D
[2 : 7; 21]		B	B	B	E	E	E	A	C	C	C	C	D	A	A	A	A	D	D
[2 : 21; 7]		B	B	E	E	E	E	C	C	C	C	D	D	A	A	A	A	A	D
[2 : 21; 21]		B	C	B	E	E	E	A	A	A	A	A	D	A	A	A	A	A	D
[2 : **7**; **7**]		A	B	E	E	E	E	A	A	A	A	C	D	A	A	A	A	D	D
[2 : **7**; **21**]		B	B	D	E	E	E	A	A	C	C	C	D	A	A	A	A	D	D
[2 : **21**; **7**]		A	B	E	E	E	E	A	A	A	C	D	D	A	A	A	A	A	D
[2 : **21**; **21**]		B	A	D	E	E	E	A	A	A	A	A	D	A	A	A	A	A	D

Note: Boldface indicates highly discriminating items; A='true dimensionality found'; B='not all items included'; C='multiple clusters'; D='dimensionality not found'; and E='all items in one subset'; and -='no outcome'.

is, item discrimination was either low or high, 7 items per trait were used, and the M2-PLM was used for generating data. This constituted a 3 (levels of mean discrimination) $\times$ 6 (correlation between traits) $\times$ 5 (lower bound c) design. In general, Table 1.4 shows that for low c-values, a high mean discrimination, and a high correlation between traits, all items were selected into one cluster (outcome Type E). As c increased, some items, depending on their discrimination and on the correlation between the latent traits, did not satisfy $H_j \geq c$ and, consequently, were not selected into this subset. When mean item discrimination was low and the correlation between latent traits was also low, with increasing c many lowly discriminating items did not satisfy $H_j \geq c$ for the first subset, but satisfied $H_j \geq c$ for the second subset. As a consequence, more clusters were found than simulated (outcome Type C). When item discrimination was high and lower bound c was also high, only the items sensitive to the same trait were collected into the same subset, thereby finding the true dimensional structure (outcome Type A). Thus, following Hemker et al. (1995) it was found that the choice of c greatly influences item selection results.

1.4.2 Comparing the DETECT and DIMTEST Test Statistics

DETECT. For data generated using the M2-PLM, Table 1.5 shows the values of $D_\alpha(\mathcal{P}^*)$ multiplied by 100; for simplicity, the result is also called $D_\alpha(\mathcal{P}^*)$. Following Zhang and Stout (1999a), $D_\alpha(\mathcal{P}^*) < 0.1$ is interpreted as essential unidimensionality and $D_\alpha(\mathcal{P}^*) > 1.0$ as sizable multidimensionality. Based on Douglas, Kim, Roussos, Stout, and Zhang (1999), $0.1 < D_\alpha(\mathcal{P}^*) < 1$ can be interpreted as moderate multidimensionality. Thus, in order to correctly interpret DETECT's results, the clustering solution as well as the value of $D_\alpha(\mathcal{P}^*)$ should be considered.

Table 1.5 shows that $D_\alpha(\mathcal{P}^*)$ is smaller as the correlation between latent traits is closer to 1.0. Also, $D_\alpha(\mathcal{P}^*)$ tends to be higher for high discrimination items, and lower when clusters contained different numbers of items (i.e., [2:7;21]). Based on the rules of thumb, for equal numbers of items per trait, for $\rho = 0.0, 0.2$, and 0.4, statistic $D_\alpha(\mathcal{P}^*)$ correctly indicated sizable multidimensionality; for $\rho = 0.6$, and 0.8, statistic $D_\alpha(\mathcal{P}^*)$ indicated moderate multidimensionality; and for $\rho = 1.0$ statistic $D_\alpha(\mathcal{P}^*)$ often indicated unidimensionality. For unequal numbers of items, statistic $D_\alpha(\mathcal{P}^*)$ indicated moderate multidimensionality, except for $\rho = 1.0$, for which $D_\alpha(\mathcal{P}^*)$ indicated unidimensionality.

Because the values of $D_\alpha(\mathcal{P}^*)$ for $\rho = 1.0$ indicate unidimensionality, the clus-

Table 1.4: *MSP Results for Different Lower Bounds c*

c	Test Composition	ρ : *0.0*	*0.2*	*0.4*	*0.6*	*0.8*	*1.0*
0.10	[2 : 7; 7]	A	D	E	E	E	E
	[2 : 7; **21**]	A	D	E	E	E	E
	[2 : **21**; **21**]	A	E	E	E	E	E
0.20	[2 : 7; 7]	A	A	D	E	E	E
	[2 : 7; **21**]	A	A	E	E	E	E
	[2 : **21**; **21**]	A	A	E	E	E	E
0.30	[2 : 7; 7]	C	C	B	D	D	D
	[2 : 7; **21**]	C	B	B	E	E	E
	[2 : **21**; **21**]	A	A	E	E	E	E
0.40	[2 : 7; 7]	C	C	C	C	C	D
	[2 : 7; **21**]	B	C	C	D	D	E
	[2 : **21**; **21**]	A	A	A	C	E	E
0.50	[2 : 7; 7]	B	C	C	C	B	B
	[2 : 7; **21**]	B	B	B	B	B	E
	[2 : **21**; **21**]	A	A	A	C	E	E

Note: Boldface indicates highly discriminating items; A='true dimensionality found'; B='not all items included'; C='multiple clusters'; D='dimensionality not found'; and E='all items in one subset'.

ters DETECT yielded for $\rho = 1.0$ and which were presented in Table 1.1 should be ignored. Unidimensionality was supported only for $\rho = 1.0$ when items discriminated highly, or when clusters contained 21 items (see Table 1.1). The results in Table 5 show that for unidimensional data $D_\alpha(\mathcal{P}^*)$ values were found as high as 0.202. This may indicate that the upper bound of 0.1 for essential unidimensionality may be too low. As one of the reviewers indicated, the rules of thumb for interpreting $D_\alpha(\mathcal{P}^*)$ may be the topic of future research.

DETECT works less well for unequal numbers of items. This result can be explained using Figure 1.3. Let [2:7;21] be the simulated dimensionality, then the direction of best measurement of the test, θ_α, lies closer to the direction of θ_2, because there are more items sensitive to this trait. As a consequence, the expected conditional covariances for items sensitive to θ_2 are closer to 0 than the expected

Table 1.5: *Results of DETECT Statistic $D_\alpha(\mathcal{P}^*)$ Using the Multidimensional Two-Parameter Logistic Model (M2-PLM) for Two Latent Traits.*

Test	ρ :					
Composition	*0.0*	*0.2*	*0.4*	*0.6*	*0.8*	*1.0*
[2 : 7; 7]	1.564	1.254	1.010	.642	.242	.186
[2 : 7; 21]	.756	.654	.535	.379	.203	.108
[2 : 21; 7]	.827	.681	.551	.380	.203	.115
[2 : 21; 21]	1.889	1.513	1.083	.731	.361	.092
[2 : 7; **7**]	1.836	1.547	1.111	.775	.441	.202
[2 : 7; **21**]	.905	.806	.621	.427	.303	.118
[2 : 21; **7**]	.851	.723	.574	.361	.208	.119
[2 : 21; **21**]	2.132	1.702	1.341	.879	.442	.090
[2 : **7; 7**]	2.137	1.750	1.266	.842	.385	.067
[2 : **7; 21**]	.978	.826	.668	.472	.259	.039
[2 : **21; 7**]	.999	.816	.653	.494	.247	.042
[2 : **21; 21**]	2.400	2.016	1.508	.967	.518	.034

Note: Boldface indicates highly discriminating items.

conditional covariances for items sensitive to θ_1 and, because of that, their dispersion also is smaller. Furthermore, item pairs from the larger clusters more easily may have negative conditional covariances, even though they are sensitive to the same latent trait, and items from different clusters may have positive expected conditional covariances, even though they are sensitive to different traits. Such incorrect sign behavior is more likely for item pairs that are sensitive to θ_2, because due to their smaller angles with θ_α they more often are on different sides of θ_α than item pairs that are sensitive to θ_1.

DIMTEST. Using DIMTEST statistic T', Table 1.6 shows that, in general, for $\rho = 1.0$ unidimensionality was found, and for $\rho \leq .8$ multidimensionality was found. These results were more pronounced for highly discriminating items. These results were not found for 14 items ([2:7;7]), maybe because T' is an asymptotic statistic.

Unidimensionality was found unexpectedly for [2:21;21] and [2:**21**;**21**] when $\rho = 0.4$. Given that AT1 must be as dimensionally distinct as possible from AT2 and PT, then both setups result in an AT1 that are sensitive to either θ_1 or θ_2, and an AT2 and a PT that is sensitive to a mixture of θ_1 and θ_2. This may result in less

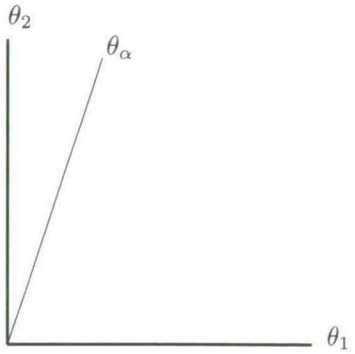


Figure 1.3: *Geometrical Representation of One Short Test (θ_1) and One Long Test (θ_2)*

power to reject the null hypothesis in these cases (also see Nandakumar & Stout, 1993), because the covariance between AT1 items will be relatively low when PT-scores are partly driven by the same latent trait. One may note that DIMTEST performs well when latent traits are represented with an unequal number of items and, to a lesser extent, with unequal discrimination. In the latter cases, AT1 and PT are in a large degree driven by distinct latent traits.

In Table 1.7, the number of times the null hypothesis was rejected using a nominal significance level of .05 is reported for five replicated data matrices. The results agree to a high degree with the findings presented in Table 1.6. For example, the results for equal numbers of items per trait are less stable than for unequal numbers of items per trait. Also, for [2:7;7] the results mainly reflect random fluctuation.

1.5 Conclusion and Discussion

Using the methods as recommended in the literature, DETECT and HCA/-CCPROX were superior to MSP in retrieving the simulated dimensional structure. Even when there was little information available to distinguish items that are sensitive to different traits (e.g., highly discriminating items, sensitive to two traits that correlated 0.8), DETECT and HCA/CCPROX retrieved the dimensional structure, but MSP failed. It may be noted, however, that traits correlating 0.8 may be indistinguishable from a substantive viewpoint, which puts the MSP result in a more positive perspective. In general, DETECT performed better than

Table 1.6: *Results of DIMTEST Statistic T' Using the Multidimensional Two-Parameter Logistic Model (M2-PLM) For Two Latent Traits*

Test	ρ					
Composition	*0.0*	*0.2*	*0.4*	*0.6*	*0.8*	*1.0*
[2 : 7; 7]	○	○	○	○	○	○
[2 : 7; 21]	•	•	•	•	○	○
[2 : 21; 7]	•	•	•	•	○	○
[2 : 21; 21]	○	•	○	•	•	○
[2 : 7; **7**]	•	○	•	•	○	○
[2 : 7; **21**]	•	•	•	○	○	○
[2 : 21; **7**]	•	•	•	•	•	○
[2 : 21; **21**]	•	•	•	○	○	○
[2 : **7; 7**]	○	•	○	○	○	○
[2 : **7; 21**]	•	•	•	•	•	○
[2 : **21; 7**]	•	•	•	•	•	○
[2 : **21; 21**]	•	•	○	•	•	○

Note: Boldface indicates highly discriminating items; • denotes significant result using .05 as significance level (i.e., multidimensionality) and ○ indicates a non-significant result (i.e., unidimensionality).

HCA/CCPROX, but in some instances, for example, when discrimination was low and tests were long, HCA/CCPROX was superior to DETECT.

Differences between DETECT and HCA/CCPROX may be due different conditional covariance estimates used in these methods. Also, DETECT's algorithm is less susceptible to locally optimal solutions than HCA/CCPROX's algorithm. In addition, in practice the researcher must choose the final HCA/CCPROX cluster solution among $J - 1$ solutions. This may be an extra source for differences.

MSP and DETECT differ in many ways. First, MSP uses normed unconditional covariances, and DETECT uses conditional covariances. Second, MSP uses a sequential clustering procedure based on several item selection criteria, and DETECT searches for the item partition that maximizes $D_\alpha(\mathcal{P})$ without additional selection criteria. Third, MSP selects items that satisfy $H_j \geq c$, where c is meant as a minimum quality criterion for item discrimination. As a result, the default setting $c = 0.3$ may not, for example, select all items driven by the same trait in a cluster and, thus, may not yield the 'true' dimensionality. Other, non-default,

Table 1.7: *Frequency (Out of 5) With Which DIMTEST Statistic T' Rejects the Null Hypothesis (5% Level), Using the Multidimensional Two-Parameter Logistic Model (M2-PLM) for Simulation.*

Test	ρ :					
Composition	*0.0*	*0.2*	*0.4*	*0.6*	*0.8*	*1.0*
[2 : 7; 7]	2	4	0	1	0	0
[2 : 7; 21]	5	5	5	4	2	0
[2 : 21; 7]	5	5	5	5	2	0
[2 : 21; 21]	5	2	4	4	3	1
[2 : 7; **7**]	0	3	1	0	1	0
[2 : 7; **21**]	5	5	5	4	2	0
[2 : 21; **7**]	5	5	5	5	3	1
[2 : 21; **21**]	4	5	5	5	3	1
[2 : **7; 7**]	2	2	2	2	0	0
[2 : **7; 21**]	5	5	5	5	5	0
[2 : **21; 7**]	5	5	5	5	5	0
[2 : **21; 21**]	5	5	5	4	4	0

Note: Boldface indicates highly discriminating items.

values of c, however, may yield the correct dimensionality. Fourth, items selected by MSP into a cluster cannot leave this cluster, and this makes MSP susceptible to locally optimal solutions. DETECT uses a genetic algorithm, which moves items back and forth until a final solution is found. These differences make the comparison of MSP and DETECT difficult.

Two remarks with respect to the DETECT and DIMTEST statistics are in order. First, it was found that the number of items in a cluster assessing one trait may influence the assessment of dimensionality; DETECT's maximum, $D_\alpha(\mathcal{P}^*)$, did not reflect the dimensionality well for data matrices which contained clusters with an unequal number of items, and DIMTEST's T' did not reflect the dimensionality well for clusters with equal numbers of items and equal average discrimination. Second, the results for $N = 200$ made clear that DETECT and DIMTEST may be more effective in larger samples (here, $N = 2,000$ was investigated).

Practical recommendations: Based on on our simulation study we found that DETECT and MSP are the most useful programs for finding unidimensional item clusters. Both methods yield a single clustering solution and provide test statistics for evaluating the quality of the cluster solution. In general, DETECT recovered the simulated dimensionality better than MSP, but DETECT needed larger samples than MSP. Further, for data sets with highly correlating latent traits DETECT forces items into clusters and DETECT seems, therefore, to be vulnerable to chance capitalization. MSP always produces the best item clustering according to the definition of a Mokken scale, and discards items not fitting well. We compared the methods as they are available to researchers. Future research may use in one selection procedure conditional covariances to find the true dimensionality of the data and a minimum item-quality criterion for only selecting practically useful items.

DIMTEST is suitable when the researcher expects his/her data to be unidimensional, but cannot be used to partition the data. DIMTEST has low power for short tests. HCA/CCPROX yields $J - 1$ cluster solutions. This forces the researcher to make a choice about the dimensionality of the test. A drawback of the method is that it does not provide the value of a quality statistic for each solution, such as $D_\alpha(\mathcal{P}^*)$ or H, on which the researcher can base his/her choice.

At the practical level, it may be noted that MSP uses a Windows interface and can be run under Windows 95 or higher. DETECT, HCA/CCPROX and DIMTEST are DOS programs.

A practical recommendation, because the methods are so different, is to use them next to one another to analyze the same data set. For example, one could use DETECT to find dimensionally distinct clusters, and use MSP to select the best discriminating items for which $H_j > c$ within these clusters. HCA/CCPROX can be informative about the process of clustering; for example, which items have most in common and are clustered first, and which are added later. DIMTEST can be used to verify unidimensionality of the clusters, especially because this method has more power than DETECT when only few items are driven by another trait than the main trait.

Chapter 2

Mokken Scale Analysis Using Hierarchical Clustering Procedures

Abstract

Mokken scale analysis can be used to assess and build unidimensional scales from responses to a multidimensional item pool. An important drawback of the Mokken scale analysis program MSP is that the sequential item selection and scale construction procedure may not find the dominant underlying dimensionality of the responses to a set of items. In this chapter, alternative hierarchical item selection procedures are investigated. The performance of four hierarchical methods and the sequential clustering method in the Mokken scale analysis context was compared using a simulation study and an empirical example. The results showed that hierarchical clustering methods can improve the search process of the dominant dimensionality of a data matrix. In particular, the complete linkage and scale linkage methods were promising in finding the dimensionality of the item response data from a set of items.

This chapter has been accepted as: Van Abswoude, A.A.H., Vermunt, J.K., Hemker, B.T. & Van der Ark, L.A. (in press). Mokken scale analysis using hierarchical clustering procedures. *Applied Psychological Measurement.*

2.1 Introduction

In the last decade, there has been an increasing interest in using nonparametric item response theory (NIRT) as a tool in test dimensionality assessment. For example, Douglas et al. (1999) investigated the dimensionality of the Law School Admission Test and Scheirs and Sijtsma (2001) investigated the dimensionality of the International Survey of Adult Crying. In NIRT it is assumed that all items are sensitive to a single latent trait (unidimensionality, UD), that item responses given this latent trait are statistically independent (local independence, LI), and for dichotomously scored items that the probability of answering an item correctly is a monotone nondecreasing function of the latent trait (monotonicity, M) (Mokken, 1971; Sijtsma, 1998). A set of items satisfying UD, M and LI is denoted as being monotonely homogeneous (MH; Mokken, 1971).

Methods that are aimed at selecting sets of items that are sensitive to one latent trait from larger item pools sensitive to multiple latent traits frequently use relaxations of the UD, M, or LI assumptions, and often focus on one or more of these weakened assumptions (e.g., Ip, 2001; Van Abswoude, Van der Ark, & Sijtsma, 2004). *HCA/CCPROX* (e.g., Roussos et al., 1998), *DIMTEST* (e.g., Nandakumar & Stout, 1993) and *DETECT* (e.g., Zhang & Stout, 1999a; 1999b) concentrate on the LI assumption. The program discussed in this chapter, *MSP* (Molenaar & Sijtsma, 2000), concentrates on the M assumption. Now the MSP procedure is discussed for dichotomously scored items.

The Mokken scale analysis method (MSA; e.g., Mokken, 1971; Molenaar & Sijtsma, 2000) and its program MSP have, apart from being generally available for applied researchers, a number of attractive properties: if the MH model is satisfied, simple sum scores may be used to stochastically order subjects on the underlying variable (i.e., Grayson, 1988; Hemker et al., 1997); it allows the user to choose only items with sufficient discrimination power into a test (Sijtsma, 1998); it is suitable in contexts where there are, compared to weak-LI methods, few items and few subjects; and the item selection procedure runs quite fast. MSP contains a sequential item selection method, which can be seen as a sequential clustering algorithm. This clustering method has, however, an important drawback: it does not always find back the correct number of underlying traits and correct dimensionality structure of the data in multi-trait situations (Van Abswoude et al., 2004).

In this study, it is investigated whether hierarchical clustering (Everitt, Landau, & Leese, 2001) algorithms can find the correct number of underlying traits better than the sequential clustering algorithm; that is, whether suboptimal so-

lutions can be prevented in multi-trait situations. We present four types of hierarchical clustering methods that can be used to create non-overlapping sets of dichotomous items, each of which satisfies certain scaling conditions. In a simulation study and an empirical example, these four methods were compared with each other and with MSP's sequential procedure in their ability to find the correct dominant underlying dimensionality structure in situations with more than one latent trait.

2.2 Mokken Scale Analysis

Mokken scale analysis is a method that may be used to select a subset of items sensitive to the same underlying dimension from a larger item pool. Similarly to factor analysis, MSA starts with a matrix containing information on the strength of the bivariate relationships between the J items under study. In factor analysis, this is either a correlation or covariance matrix. MSA uses a matrix with H-coefficients (Loevinger, 1948; Mokken, 1971). Let F_{jk} represent the observed number of Guttman (1950) errors for item pair (j, k) and E_{jk} the expected number of Guttman errors under the null model of marginal independence. If item j is easier than item k (i.e., more correct answers), a Guttman error occurs when the more difficult item k is answered correctly and the easier item j is answered incorrectly. The pairwise H_{jk} for item pair (j, k) is defined as

$$H_{jk} = 1 - \frac{F_{jk}}{E_{jk}} = \frac{E_{jk} - F_{jk}}{E_{jk}}.$$

Let X_j be a binary random variable, denoting an individual's score on item j $(j = 1, ..., J)$. An item score may take on the value 0 or 1, and let N be the sample size. Furthermore, let $\pi_j = P(X_j = 1)$, $1 - \pi_j = P(X_j = 0)$, and $\pi_{jk} = P(X_j = 1, X_k = 1)$. Items are ordered such that $\pi_j > \pi_k$, for all $j > k$. Noting that $F_{jk}/N = \pi_k - \pi_{jk}$ and $E_{jk}/N = (1 - \pi_j)\pi_k$, H_{jk} can also be written as

$$\begin{aligned} H_{jk} &= \frac{(1 - \pi_j)\pi_k - (\pi_k - \pi_{jk})}{(1 - \pi_j)\pi_k} \\ &= \frac{\pi_{jk} - \pi_j\pi_k}{(1 - \pi_j)\pi_k}. \end{aligned}$$

The numerator of H_{jk} equals the covariance between binary variables, and the denominator the maximum positive value of the covariance given the marginal distributions of the item scores. Let $\text{cor}(X_j, X_k)$ denote the correlation between the responses on items j and k and $\text{cor}(X_j, X_k)_{max}$ the maximum correlation

given the marginal distributions of items j and k. Then, H_{jk} can also be defined as

$$H_{jk} = \frac{\text{cor}(X_j, X_k)}{\text{cor}(X_j, X_k)_{\max}}.$$

Thus, the H_{jk} can be seen as a normed correlation coefficient; that is, a measure that takes into account that the correlation cannot reach the maximum value of 1 for items with different proportions correct scores π_j and π_k.

From the pairwise H_{jk}, one can derive H_j for each item belonging to a scale, as well as the overall scale H. H_j indicates how well item j fits into a scale and H indicates the strength of the scale. High values of H indicate a more correct ordering (i.e., fewer Guttman errors) of subjects on the latent trait.

The H_j for item j can be defined as follows:

$$H_j = 1 - \frac{\sum_{k \neq j} F_{jk}}{\sum_{k \neq j} E_{jk}} = \frac{\sum_{k \neq j} (E_{jk} - F_{jk})}{\sum_{k \neq j} E_{jk}}$$

as in Mokken (1971, p. 150). It can also be written as a weighted mean of H_{jk} coefficients, with weights equal to E_{jk}; that is,

$$H_j = \frac{\sum_{k \neq j} E_{jk} H_{jk}}{\sum_{k \neq j} E_{jk}}. \tag{2.1}$$

The scale H is defined as

$$H = 1 - \frac{\sum_j \sum_{k \neq j} F_{jk}}{\sum_j \sum_{k \neq j} E_{jk}} = \frac{\sum_j \sum_{k \neq j} (E_{jk} - F_{jk})}{\sum_j \sum_{k \neq j} E_{jk}}$$

It can easily be verified that H can also be defined as a weighted mean of pairwise H_{jk} or item H_j,

$$H = \frac{\sum_j \sum_{k \neq j} E_{jk} H_{jk}}{\sum_j \sum_{k \neq j} E_{jk}} = \frac{\sum_j \left(\sum_{k \neq j} E_{jk} \right) H_j}{\sum_j \sum_{k \neq j} E_{jk}}, \tag{2.2}$$

where the weights are equal to E_{jk} and $\sum_{k \neq j} E_{jk}$, respectively.

Let $\text{cov}(X_j X_k)$ denote the covariance of variables j and k. A set of J items are called a *Mokken Scale* (Mokken, 1971, p. 184) if all items satisfy the following two conditions,

Condition 1 $\text{cov}(X_j X_k) > 0$, for all $j \neq k$, and

Condition 2 $H_j \geq c$, for all j, where c is a user defined constant between 0 and 1.

The first condition follows from the UD, LI and M assumptions of the MH model (Mokken, 1971, p. 149; also see, Holland & Rosenbaum, 1986). This condition can also be restated as $\text{cor}(X_j X_k) > 0$ or $H_{jk} > 0$. The second condition serves the practical purpose that only items with sufficient discrimination power are accepted into a scale (e.g., Sijtsma, 1998). More generally stated; the higher c, the more accurate the ordering of persons on the underlying trait θ. Mokken chose $c = 0.3$ in Condition 2, which over the years has been shown to be a good rule of thumb in measuring a single underlying dimension. Mokken also provided rules of thumb for interpreting the strength of a scale (Mokken, 1971, p. 185). Hemker et al. (1995) provided suitable values for c for finding the simulated dimensionality structure of item pools having different item characteristics.

For a set of items satisfying a Mokken scale, the following inequalities hold:

$$0 < \min(H_{jk}) \leq \min(H_j) \leq H \leq \max(H_j) \leq \max(H_{jk}) \leq 1 \qquad (2.3)$$

(see, Hemker et al., 1995; Mokken, 1971). This means, for example, that the lowest H_j in the scale is at least equal to the lowest H_{jk}. When searching for scales satisfying the Mokken scale conditions using a clustering procedure (sequential, hierarchical, or otherwise), one will generally see that H and also the H_j decrease when the number of items in the scale increases.

The H coefficient, and MSA in general, has been extended to polytomous items (Molenaar, 1982; 1991). The H coefficient for polytomous items also is the normed covariance between item pairs, and the interpretation of the coefficient remains the same (Hemker et al., 1995). Although the hierarchical procedures are discussed only for dichotomous items, they can be readily applied to item sets consisting of polytomous items.

2.3 Sequential Clustering

One way to find a Mokken scale is by checking whether an a priori selected set of items satisfies the two conditions described above. Items that do not fulfill these conditions should be removed. Another, more exploratory, approach involves applying a stepwise item selection procedure rather than specifying an a priori selected set. Such a stepwise procedure has been implemented in the program *MSP* (Molenaar & Sijtsma, 2000).

The sequential item selection procedure works as follows:

Step 1 Select the two items with the highest significantly positive H_{jk} in the sample.

Step 2 Compute H for all remaining items with respect to the already selected items, and select the item with the highest H that satisfies Conditions 1 and 2.

Step 3 Repeat Step 2 until no items remain that satisfy Conditions 1 and 2. If items remain, go to Step 1 to form another scale using the remaining items. If no more items remain, the entire procedure stops.

The default value for c is 0.3. The default α value used in the one-sided significance tests for H_{jk} and H_j is 0.05. In order to reduce the risk of capitalizing on chance, the level of significance is adjusted using a Bonferroni correction at each step. It is possible to change the default values for c and α, as well as to use other starting sets in Step 1. In the situation of a tie in Step 1 or 2, the item pair with the lowest π_j is selected.

It should be noted that MSP's item selection algorithm can also be regarded as a sequential clustering algorithm. The objects to be clustered are the items, and the matrix with the H_{jk} serves as proximity matrix between the items. The H serves as a proximity measure between a set of items forming a cluster or scale and the remaining single items that could be added to the cluster that is being built. Clusters are formed sequentially; that is, only after one cluster has been formed, item selection for a second cluster starts. The end result is a set of clusters, each consisting of two or more items, and each forming a Mokken scale. Items that do not satisfy the scaling conditions for any scale in Step 2 are not entered in any cluster. In general, the higher the discrimination of the items in an item pool, the less nonscalable items there are.

A drawback of the sequential clustering procedure is, however, that it may yield a suboptimal solution in the sense that the true dimensionality of the data may not be found (Van Abswoude et al., 2004). Let us illustrate this phenomenon by means of a small example consisting of the responses on six items, denoted Item1, $\cdots$, Item6, and two independent latent traits, θ_1 and θ_2. Assume that Item1 and Item2 are strongly related to θ_1, Item3 is weakly related to θ_1 and strongly related to θ_2, and Item4, $\cdots$, Item6 are moderately related to θ_2 but not related to θ_1. The H_{jk} matrix and π_j-values of these six items are presented in Table 2.1.

Using MSP's defaults settings, Step 1 will yield item pair Item1-Item2 as starting pair. Subsequently, Item3 will be added to that cluster. The final first scale will consists of Item1, Item2 and Item3, with $H = 0.46$. The second scale will contain the remaining three items Item4, Item 5 and Item6, with $H = 0.65$. This two-cluster solution is suboptimal because there exists a solution that better re-

Table 2.1: *Lower Diagonal of H_{jk}-Matrix and Item Popularities π_j for Small Simulated Example*

	Item1	Item2	Item3	Item4	Item5	Item6
Item1						
Item2	0.87					
Item3	0.32	0.31				
Item4	0.01	-0.02	0.57			
Item5	0.06	0.04	0.56	0.48		
Item6	0.00	0.00	0.73	0.79	0.84	
π_j	0.68	0.38	0.50	0.75	0.62	0.27

flects the true dimensionality of the responses to the six items. This solution consists of two clusters containing Item1-Item2 and Item3, Item4, Item5, Item6, respectively. The H values for these scales are 0.87 and 0.63[1].

The problem of the sequential clustering procedure is, of course, that the starting set determines the solution. Once an item is selected in a cluster it remains there, even if it fits better into a cluster that is formed later. In our example, the better solution would be obtained if the method would start with the second highest H_{jk} and then continue using MSP's default settings. The MSP software offers the possibility to specify other starting sets or extent the search procedure to allow for overlapping clusters, which means that it would have been possible to find the better solution. Unexperienced users may, however, not override the default settings and may therefore end up with suboptimal solutions.

2.4 Hierarchical Clustering

A possible solution to the problem associated with MSP's sequential item clustering method may be to switch to another type of clustering method. Hierarchical clustering analysis (HCA; e.g., Everitt et al., 2001) may be a useful method to find sets of items that form a Mokken scale.

[1]We also calculated the correlation between the sum scores on subsets of these items; these sets are item pair Item1-Item2 (set 1), Item3 (set 2), and Item4, Item5 and Item6 (set 3). The Pearson product moment correlation coefficient (ρ) between the sum scores for set 1 and 2 equals 0.272, for set 1 and 3 ρ=0.022, and for set 2 and 3 ρ=0.540. Thus, these correlations indicate that there is no linear relationship between the sets 1 and 3, and that Item3 fits both in set 1 and in 3, but that the linear relationship with the items in set 3 is highest.

Starting point of HCA is a data matrix containing proximities between separate objects. In our case, the separate objects are items and their proximities are their pairwise H_{jk}. At each hierarchical step, the two objects that are most similar are joined. A joined pair is also called an object or cluster. This means that at any hierarchical step two single items may be clustered to form one new cluster, a single item may be added to an existing cluster of items, or two clusters may be combined into a single larger cluster. This process continues until some previously stated criterion (i.e., a stopping rule) is met or until all items are in one single cluster.

Hierarchical clustering has been used before for dimensionality assessment in a classical test theory context (e.g., Reveille, 1979; Schweizer, 1991; Hunter, 1973; Bacon, 2001) as well as in a NIRT modelling context (Roussos et al., 1998). These studies have in common with each other and with our study that they use variants of product moment correlations coefficients as proximity metrics. For example, Reveille (1979) used Cronbach's alpha and the worst split-half (beta) coefficient to form clusters having high internal consistency reliability, and Roussos et al. (1998) used conditional correlations and conditional covariances to obtain clusters that are weakly locally independent. These studies differ from our study in the specification of minimal requirements on the clusters constructed with HCA, meaning that two items that correlate negatively could end up in the same cluster. New in our hierarchical clustering procedures is that clusters constructed with the HCA need to satisfy some minimal requirements, the Mokken scaling conditions.

Let O_v denote an object consisting of one or more items. The number of items in O_v is denoted by J_v, and the proximity between objects O_v and O_w by $H_{O_vO_w}$. The way the proximities are calculated will be explained later on. Hierarchical clustering of items uses the following steps:

Step 1 Join the two items j and k with the highest H_{jk} that satisfy the scaling conditions.

Step 2 Compute the $H_{O_vO_w}$ between all object pairs O_v and O_w, and join the object pair with the highest $H_{O_vO_w}$ as long as the stopping rule is not satisfied.

Step 3 Repeat Step 2, until no combination of two objects remain that satisfy the stopping rule.

Before describing the various HCA methods in more detail, let us return to the small example introduced in the previous section. When using HCA, individual items are not allocated to one cluster at the time, but multiple clusters may be

formed simultaneously. This means that HCA should be able to allocate X_3 to the cluster it fits best in terms of dimensionality; that is, to the second in stead of the first cluster.

Proximities

In this study, the performance is investigated of four types of agglomerative HCA methods: *complete linkage*, *average linkage*, *within-groups linkage*, and *scale linkage*. Although each of the methods uses the same three steps, they differ with respect to the definition of $H_{O_vO_w}$, or the proximity between clusters of items. The first three methods are available in the clustering routines of most statistical packages, like, for instance, SPSS (1998). The scale linkage method is a nonstandard method that is especially developed for the Mokken scaling problem.

A complete linkage method is obtained by defining the proximity between clusters as,

$$H_{O_vO_w}^{complete} = \min(H_{jk}), \text{ where } j\epsilon O_v \text{ and } k\epsilon O_w.$$

In other words, at each step those two objects O_v and O_w are joined for which the least similar pair of items has the highest proximity. Complete linkage is also known as the furthest neighbor method.

Average linkage, also known as the unweighted pair-group method of averages (Sokal & Michener, 1958) or between-groups linkage (Everitt et al., 2001), defines the proximity between objects as

$$H_{O_vO_w}^{average} = \frac{\sum_{j\epsilon O_v}\sum_{k\epsilon O_w} H_{jk}.}{J_vJ_w}.$$

As can be seen, $H_{O_vO_w}^{average}$ is the unweighted average of the bivariate H_{jk} *between* the items in object v and the items in object w. This measure of proximity therefore reflects the average distance of items belonging to different clusters.

Within-groups linkage defines the proximity of two objects O_v and O_w as the unweighted average of the H_{jk} of all items *within* O_v or O_w. In other words,

$$H_{O_vO_w}^{within} = \frac{\sum_j\sum_{k\neq j} H_{jk}}{(J_v+J_w)(J_v+J_w-1)}, \text{ where } \{j,k\}\epsilon O_v \cup O_w.$$

The fourth method, scale linkage is based on the scale H of the possible new object that is obtained by joining two objects; that is,

$$H_{O_vO_w}^{scale} = \frac{\sum_j\sum_{k\neq j} E_{jk}H_{jk}}{\sum_j\sum_{k\neq j} E_{jk}}, \text{ where } \{j,k\}\epsilon O_v \cup O_w.$$

Scale linkage may be regarded as the a hierarchical clustering variant of the sequential clustering procedure of the MSP package: that is, those objects are joined that together result into the largest possible scale H.

In order to be able to define stopping rules on the basis of the four types of proximity measures, it is important to get some idea on their meaning in the context of Mokken scaling. The first and last proximity measures have a direct interpretation in the context of Mokken scaling: $H_{O_vO_w}^{complete}$ is directly related to the minimal requirement for items belonging to the same scale (Condition 1; $H_{jk} > 0$), while $H_{O_vO_w}^{scale}$ is the overall summary measure of the quality of a scale, see Equation 2.2. The other two measures can be seen as proxies for Mokken scaling measures. As can be seen, $H_{O_vO_w}^{within}$ is an *unweighted* average of H_{jk} and can, therefore, be interpreted as an unweighted approximation of the overall scale H, which was a *weighted* coefficient of the H_{ik}s. Similarly, $H_{O_vO_w}^{average}$ can be seen as an unweighted proxy for H_j coefficients. More precisely, for each item belonging to object v, an unweighted item H_j could be computed indicating how well it fits into object w, $H_j^{O_w} = \sum_{k \epsilon O_w} H_{jk}/J_w$. The measure $H_{O_vO_w}^{average}$ equals the average of these $H_j^{O_w}$s: $H_{O_vO_w}^{average} = \sum_{j \epsilon O_v} H_j^{O_v}/J_v$.

Stopping rule

The most natural point to stop a hierarchical clustering process is when the largest proximity drops below some minimum value; that is, stop the process of joining objects if $\max(H_{O_vO_w}) < c$. When applying the HCA in this study, it is important to choose the stopping rule in such way that it leads to clusters that satisfy the same scaling conditions as the ones required with sequential clustering[2].

Defining a stopping rule based on $H_{O_vO_w}^{complete}$ is not straightforward since there is no direct relationship between the minimal H_{jk} of a scale and the overall homogeneity of the items (H_js). From Conditions 1 and 2 and Equation 2.3, the requirement that $0 < H_{O_vO_w}^{complete} \leq c$ can be derived. It, therefore, seems reasonable to set the minimum value of $H_{O_vO_w}^{complete}$ somewhat larger than zero, say $c^{complete} = 0.10$, to increase the quality of measurement.

Using the same line of reasoning, hypothesis about the value of the stopping rules for the other methods can be derived using Conditions 1 and 2 and

[2]The choice of a stopping rule is a function of the research objective. For some objectives it may be preferable to join subsets of items sensitive to highly correlated traits since joining also increases test reliability, and for other objectives joining is undesirable. In MSA, constant c was incorporated (see, scaling Condition 2) to allow such research objectives to be incorporated in scale construction. The proposed stopping rules should be regarded as versions of this scaling condition that are adapted to the various HCA methods discussed in this chapter.

Equation 2.3. The overall scale quality, $H^{scale}_{O_vO_w}$, and the unweighted overall scale quality, $H^{within}_{O_vO_w}$, should be at least as large as the value one would use for c in the sequential procedure (because $\min[H_j] \leq H$), and maybe somewhat larger, say 0.40 . Because of its relationship to the item H_j, the same minimum value seems to be reasonable for $H^{average}_{O_vO_w}$ (i.e., $c^{average} = c = 0.30$).

Rather than using *method-specific stopping rules*, it is also possible to use a more *general rule stopping rule*. This means that the methods still maximize different proximities at each clustering step, but for each method the process of clustering stops when the same condition is no longer satisfied. As a general stopping rule, $H_j < c$, which is directly related to the Mokken scale conditions, can be used. In the simulation study, both method-specific and general stopping rules will be used.

Mokken scale conditions

In the sequential clustering algorithm, the conditions that define a Mokken scale are used as stopping rule; that is, if the conditions are no longer fulfilled the algorithm starts forming the next scale when any scalable items were left. The scales that are formed by means of HCA should satisfy the same Mokken scale conditions. The application of the conditions is, however, much more complicated within these clustering procedures.

Let us first translate the two Mokken scale conditions in such a way that they could be applied in HCA. For all objects O_v one should check whether the following conditions hold:

$$\begin{aligned} H_{jk} &> 0, \text{for all} \{j,k\} \epsilon O_v, \\ H_j &\geq c, \text{for all items } j, \text{where } j \epsilon O_v. \end{aligned}$$

The above conditions could, as in the sequential clustering procedure, be checked within the clustering process, which amounts to using an additional stopping rule. This seems to be too strict for our HCA methods. Such as strategy could, for instance, impair the clustering of two objects just because a single item does not satisfy the Mokken scaling conditions while the remaining, and possibly large set of items do satisfy the conditions. As a result, many small clusters are found.

Alternatively, the Mokken scale conditions can be checked after clusters have been formed. A stopping rule is used to stop the clustering process at a certain number of clusters and an additional step is added to the HCA method in which misfitting items are deleted from the scales. If there is more than one misfitting

item in a scale, the items are deleted sequentially, where the worst item is deleted first. The worst misfitting item is that item j with the lowest H_j amongst the items with negative H_{jk}s, or the item with the lowest H_j when negative H_{jk}s do not exist. This is the procedure followed in the simulation study.

2.5 Simulation Study

2.5.1 Description of the design

A simulation study was conducted in order to evaluate the performance of five clustering procedures: sequential, complete linkage, average linkage, within-groups linkage, and scale linkage. We used method-specific stopping rules for each clustering method. For sequential clustering, $c = 0.3$ and $c = 0$ were used. These conditions reflect two possible ways to conduct scale construction: quite restrictive ($c = 0.3$), as in MSP's default; and lowly restrictive ($c = 0.0$), as in Roussos et al. (1998) or in Zhang and Stout (1999a). In hierarchical clustering, the four method-specific stopping rules had the values $c^{complete} = 0.1$, $c^{average} = 0.3$, and $c^{within} = c^{scale} = 0.4$. To facilitate the comparison between sequential clustering and hierarchical clustering, the performance of $c = 0.3$ (for H_j) as a general stopping rule was investigated for the four hierarchical clustering methods.

Apart from the type of algorithm and the type of stopping rule, also the data structure was varied. More precisely, the size of the correlations between the traits (5 conditions) and the number of items per dimension (3 conditions) were varied. The number of latent traits was always three. Because we were not interested in sampling fluctuation issues, extremely large sample sizes ($N = 100,000$), which approximate population data, were used. For a few cells, however, also the performance of the methods using a sample of $N = 200$ was investigated.

We used three conditions with identical correlations for each pairs of traits ($\rho_{12} = \rho_{23} = \rho_{13}$) and two conditions with unequal correlations. The three conditions with identical correlations were 0.1, 0.4, and 0.7, representing weak, modest, and strong correlations, respectively. Because in most practical test applications latent traits show dependencies (McDonald, 2000), a no-dependency condition ($\rho = 0.0$) was not included. The lower ρ, the easier it should be to find the correct dimensionality. In the two situations with unequal correlations, $\rho_{12} = \rho_{13} = 0.20$ and $\rho_{23} = 0.60$, and $\rho_{12} = \rho_{13} = 0.40$ and $\rho_{23} = 0.60$ were used. Finding the correct dimensionality may be more difficult for unequal correlations than for equal correlations between latent traits. These different correlations are realistic in practical test applications.

The number of items per dimension were either 5 or 10; that is, each of the three traits may be represented by a small (5) or a large (10) number of items. Notation $[D : J_1; J_2; ...; J_D]$ is used to reflect the structure of an item pool, where D equals the *simulated* number of traits and $J_d (d = 1, ..., D)$ equals the number of items sensitive to each trait. The three conditions used were $[3 : 5; 5; 5]$, $[3 : 10; 10; 10]$, and $[3 : 5; 5; 10]$, where the latter condition represents a situation with an unequal number of items per dimension. We included the situation with unequal item numbers because Van Abswoude et al. (2004) encountered that the non-hierarchical clustering procedure DETECT (Zhang & Stout, 1999a) is less successful under such a condition.

The performance of the various procedures (methods and stopping rules) were evaluated by means of two criteria. The first criterion is whether the item selection procedure retrieves the true dimensionality of the problem or, in other words, whether items sensitive to the same latent trait are assigned to the same cluster. The second performance criterion is the overall fitness of the partition. This was quantified as the number of items that must be discarded because of misfit with respect to the two Mokken scaling conditions.

2.5.2 True model of the item responses

The model used for generating item responses was a three-dimensional version of the five-parameter acceleration model (M5-PAM; Van Abswoude et al., 2004; also see Sijtsma & Van der Ark, 2001). We used this model because it has the flexibility to represent a large variety of nondecreasing item response functions (IRFs) that may be typical for nonparametric IRT models. In our M5-PAM model, the probability that subject i answers item j correctly given her values on the three latent traits equals

$$P(X_{ij} = 1|\theta_{i1}, \theta_{i2}, \theta_{i3}) = \gamma_j^{lo} + (\gamma_j^{up} - \gamma_j^{lo}) \left[\frac{\exp\left(\sum_{d=1}^{3} 1.7\alpha_{jd}\theta_{id} + \delta_j\right)}{1 + \exp\left(\sum_{d=1}^{3} 1.7\alpha_{jd}\theta_{id} + \delta_j\right)} \right]^{\xi_j}.$$

Here, α_{jd} is the discrimination parameter of item j on trait d, and δ_j is frequently referred to as the difficulty parameter of item j. In M5-PAM, the slope and the location of the IRFs does not only depend on the α_{jd} and δ_j parameters, but also on γ_j^{lo}, γ_j^{up} and ξ_j, which are the lower asymptote, the upper asymptote, and the acceleration parameter of the IRF of item j, respectively. The last parameter makes an IRF asymmetrical (also see, Samejima, 2000).

Some items were assumed to be sensitive to a single latent trait (Zhang & Stout, 1999a called this condition simple structure) while others were assumed to

Table 2.2: *Number of Clusters and Number of Items per Cluster Obtained With Sequential Clustering for Simulated Data Sets*

	ρ :				
Test Composition	0.1	0.4	0.7	0.2/0.6	0.4/0.6
			$c = 0.3$		
$[3:5;5;5]$	[3:5;5;5]	[2:11;4]	[1:15]	[2:11;4]	[2:11;4]
$[3:5;5;10]$	[3:5;5;10]	[2:16;4]	[1:20]	[2:16;4]	[1:19]†
$[3:10;10;10]$	[3:10;10;10]	[1:30]	[1:30]	[2:21;9]	[1:30]
			$c = 0.0$		
$[3:5;5;5]$	[1:15]	[1:15]	[1:15]	[1:15]	[1:15]
$[3:5;5;10]$	[1:20]	[1:20]	[1:20]	[1:20]	[1:20]
$[3:10;10;10]$	[1:30]	[1:30]	[1:30]	[1:30]	[1:30]

Note. †, one item excluded because of misfit with scale conditions.

be strongly sensitive to one trait and less strong to the other two traits. Values for θ_{id} were drawn from a trivariate normal distribution with means of zero and correlations depending on the condition. The values of the other parameters were fixed: α_{jd} ranged between 1.50 and 2.25 for dominant traits and was set to 0.25 or 0.0 for non-dominant traits, δ_j ranged from -1.5 to 1.5, γ_j^{lo} from 0.0 to 0.1, γ_j^{up} from 0.9 to 1.0, and ξ_j from 0.5 to 3.0. Item parameters were fixed in order to obtain items that are representative for true test situations. Another option would have been to generate parameters from certain distributions, but using this approach it is difficult to obtain representative items.

2.5.3 Results

Sequential clustering

Table 2.2 shows the detected dimensionality when using sequential clustering. The first column reports the true test composition. The remaining columns report the retrieved dimensionality for the five conditions related to the correlations between latent traits. Notation $[K : J_1; J_2; ...; J_K]$ is used to represent the number of detected clusters (K) and the number of items per cluster (J_k).

The higher the correlations between traits ($\rho = 0.4$ or 0.7), the fewer the number of clusters detected. In these cases, items sensitive to different latent traits were collected into one cluster. Clustering simulated data based on traits with

varying correlations showed the same trend; that is, items that were sensitive to highly correlating traits were collected into one large cluster. Varying the number of items lead to approximately the same number of clusters. Changing the scaling conditions from $c = 0.3$ to $c = 0.0$ resulted into a single cluster containing all items.

Hemker et al. (1995) and Van Abswoude et al. (2004) found similar results for the sequential clustering procedure. Increasing the correlations between latent traits, or decreasing c in Criterion 2, means that an increasing number of items satisfy the scaling conditions of the first cluster, and consequently fewer items remain to be collected in the second or third cluster. If the scaling conditions were restricted (i.e., increase c to 0.3), depending on the type of data, sequential clustering could find the true dimensionality.

Given that K clusters are found (e.g., also when $K \neq D$), a solution may still be acceptable in the sense that items that are sensitive to the same latent trait are in the same cluster. As can be seen in Table 2.2, sequential clustering did not always lead to acceptable solutions in this sense. The result $[2 : 11; 4]$ where $[3 : 5; 5; 5]$ and $\rho = 0.4$ was simulated, for example, not only means that items sensitive to two different traits (θ_2 and θ_3) were selected into cluster 1, but also an item that mainly was sensitive to θ_1 was selected into cluster 1. The remaining items were selected into cluster 2. Because objects were clustered one-by-one, the cluster that was formed first may be over-represented compared to the cluster that was formed second.

In the main study, to get a global idea about the effect of ρ on the methods' success in dimensionality assessment some (indicative) values were used. For a few design conditions, it was investigated in more detail upto which values of ρ correct recovery stopped. In particular, for $\rho = 0.1$, 0.2 and 0.3 using a simulated data set having five item responses based on three latent traits and $c = 0.3$. For $\rho = 0.1$ the cluster solution was [3:5;5;5], for $\rho = 0.2$ the cluster solution was [3:5;4;5], and for $\rho = 0.3$ the cluster solution was [3:7/4/4]. These additional results suggest that for $c = 0.3$ the true dimensionality is recovered upto a value of ρ that lies between 0.1 and 0.2.

Hierarchical Clustering

Three types of results are presented for the hierarchical clustering methods. First, to compare the performance of the five clustering methods, the number of clusters found and the number of items that were deleted sequentially due to misfit to the Mokken scaling conditions (denoted as: number of misfitting items) are reported.

We used the hypothesized c-values for the stopping rules. In general a large sample was used, but for a few cells also the stability of the methods using small samples was investigated. Second, to find out whether the correct rules of thumb for c were used, a different perspective was adopted: For what ranges of the value of c would the simulated number of latent traits have been recovered? These ranges may give us more insight into the functioning of the four new proximities for different sets of items (for similar ranges for H_j using sequential clustering, see Hemker et al., 1995). Finally, it was investigated whether the graphical plots of the proximities at each hierarchical step contain information on the actual dimensionality.

Table 2.3 reports the number of clusters and the number of misfitting items (in parenthesis). The first part shows the results using the general stopping rule $H_j < 0.3$ (all methods), and the second part shows the results of each hierarchical clustering method in combination with its method-specific stopping rule. The specific number of items in each cluster is not reported for clarity of Table 2.3. Before discussing the effects of the general and method-specific stopping rules, some general results will be described.

As far as the number of clusters are concerned, the hierarchical clustering methods showed approximately the same pattern as sequential clustering: the number of clusters decreased when the correlations between latent traits increased and the restrictiveness of clustering decreased. Unlike sequential clustering, the number of clusters did in some instances change between conditions with different numbers of items per latent trait: the number of clusters decreased when the number of items decreased. This result, which is caused by minor differences in the parameter values when 5 or 10 items per trait were simulated, was found for some hierarchical methods (i.e., average linkage and within-groups linkage) using the proposed stopping rules.

Because hierarchical clustering forms clusters simultaneously instead of sequentially, HCA should collect items sensitive to the same trait into the same cluster. One can observe in Table 2.2 that especially the 2-cluster solutions of sequential were suboptimal. The results using HCA are better; that is, $[2:10;5]$ for $J = 15$, $[2:15;5]$ for $J = 20$ and $[2:20;10]$ for $J = 30$ (the number of items per cluster are not shown in Table 2.3). In HCA items sensitive to the same dominant trait were not joined into different clusters.

The number of misfitting items as reported in Table 2.3 gives an indication of the quality of a solution. As can be seen, most misfitting items were found when ρ was low or moderate and for a small number of clusters. Misfitting items could, of course, be prevented by increasing the restrictiveness of item clustering. However, the absence of misfitting items does not indicate that the true dimensionality was

Table 2.3: *Number of Clusters and Number of Misfitting Items (in parenthesis) Obtained With Hierarchical Clustering for Simulated Data Sets*

	ρ :				
Test Composition	0.1	0.4	0.7	0.2/0.6	0.4/0.6
General stopping rule					
	$H_j < 0.3$ (all methods)				
$[3:5;5;5]$	3	2	1	2	2
$[3:5;5;10]$	3	2	1	2	2
$[3:10;10;10]$	3	1	1	2	1
Method-specific stopping rules					
	$H_{O_vOw}^{Complete} < 0.1$				
$[3:5;5;5]$	3	1(1)	1	2	1(1)
$[3:5;5;10]$	3	1(1)	1	2	1(1)
$[3:10;10;10]$	3	1	1	2	1
	$H_{O_vOw}^{Average} < 0.3$				
$[3:5;5;5]$	5	6	6	5	5
$[3:5;5;10]$	5	5	4	4	6
$[3:10;10;10]$	4	4	4	4	4
	$H_{O_vOw}^{Within} < 0.4$				
$[3:5;5;5]$	3	2	1	2	1(1)
$[3:5;5;10]$	2(5)	1(1)	1	1(4)	1(1)
$[3:10;10;10]$	3	1	1	1(2)	1
	$H_{O_vOw}^{Scale} < 0.4$				
$[3:5;5;5]$	3	2	1	2	2
$[3:5;5;10]$	3	1(1)	1	2	1(1)
$[3:10;10;10]$	3	2	1	2	1

found, since it may be a solution with too many clusters.

The *general stopping rule* ($H_j < 0.3$) was applied to all HCA methods. The results were the same for all of these methods (results are presented only once in Table 2.3). This means that the path item clustering initially took had no effect on the final solution. When comparing the results of the general stopping rule with sequential clustering, one may observe that the number of clusters were the same. As was explained before, the specific items in each cluster were somewhat dissimilar. The improvement of hierarchical clustering over sequential clustering lies in the fact that using HCA items sensitive to the same latent trait were collected into the same cluster, whereas using sequential clustering this was not always the case.

The four *HCA methods* in combination with their *method-specific stopping rules* were not equally successful in finding the true dimensionality. In complete linkage, using the method-specific stopping rule, for conditions with modest (i.e. $\rho = 0.4$) and high (i.e., $\rho = 0.7$) correlations between latent traits the number of clusters found was less than the true number of latent traits. This suggests that the method-specific stopping rule (i.e., the minimum of H_{jk}) for complete linkage was too low to find the correct dimensionality. In within-groups linkage, and to a lesser extent in scale linkage, which both use functions of the scale H, similar trends can be observed. Scale linkage performed better than within-groups linkage (see Table 2.3) because in scale linkage a proximity measure was used that corrects for variations in item difficulties. In within-groups linkage, especially in the conditions with 10 items per latent trait, too few clusters were found. Using the average linkage method with the average linkage criterion more clusters than the true number were found. For many item pools, this meant that a different cluster was generated for each possible combination of traits: that is, sets of items sensitive to either the same latent trait or a same combination of latent traits were collected in one cluster. Seemingly, average linkage does not serve as good proxy for H_j. Correct recovery stopped at the following values of ρ: *general stopping rule* between 0.2 and 0.3; complete linkage between 0.2 and 0.3; within-groups linkage between 0.2 and 0.3; and scale linkage between 0.3 and 0.4 (not shown in Table 2.3). Thus, when increasing ρ in smaller steps the scale linkage method performed best. To sum up, scale linkage seems to be the only method that in combination with the method-specific stopping rule performs better than the general stopping rule.

These results may not hold up for small samples. Therefore, the success of the methods and the specific stopping rules was investigated for $N = 200$. Table 2.4 shows results of ten replications for small (i.e., $N = 200$) and large (i.e., $N =$

100,000) sample sizes. The table presents two types of results. First, for each method the number of times the correct three cluster solution was obtained using the four hierarchical clustering methods is presented. Alternatively, the average number of clusters over 10 replications could have been presented. This, however, is not informative because it says nothing about whether and how often the correct number of clusters is obtained and whether this number of clusters corresponded with the true underlying dimensionality. Secondly, the standard error (SE) of the proximities when the correct clustering was found using the method-specific stopping rule is presented for small and large sample sizes. The SE is calculated using the proximity value obtained with $N = 100,000$ as true score. Let Y_r denote the proximity value of a HCA method for replication r. In addition, let μ denote the average of the proximity for $N = 100,000$ (it is referred to in Greek notation because it approaches the population value). Then, the SE equals,

$$SE = \sqrt{\frac{1}{10}\sum_{r=1}^{10}(Y_r - \mu)^2}. \tag{2.4}$$

Furthermore, the results obtained for $N = 100,000$ response patterns presented earlier (see, Table 2.3) suggested that the correlation conditions $\rho = 0.7, 0.2/0.6$ and $0.4/0.6$ may not yield the true dimensionality. For that reason, the results for the conditions $\rho = 0.1, 0.2, 0.3$ and 0.4 that may find the true dimensionality were presented.

The results in Table 2.4 show that for small samples the methods using the method-specific stopping rules worked less well than than using large samples. As indicated by the standard errors of the proximities (see Table 2.4), the proximities in some small samples fell below and in others fell above the value of the predefined stopping rules. Thus, the stopping rules were not equally suitable for large and small sample sizes.

In the second type of results a different perspective is adopted: instead of fixing c, it is searched for what *ranges of* c the correct dimensionality for the method-specific stopping rules is obtained. These ranges are presented in Table 2.5. All methods were able to find the correct number of clusters and joined the correct items. In general, the higher ρ, the more restrictive the scaling conditions need to be in order to find the true dimensionality.

The range of c strongly changes with the proximity used for clustering. Using H_{jk} as proximity (in complete linkage) is very attractive because a large range of c-values leads to the correct result. Table 2.5 provides a confirmation that hypothesized values of c were too low for complete linkage, within-groups linkage and scale linkage, and too high for average linkage to find the correct solution.

Table 2.4: *Numbers of Correct Clusters (#) and the Proximities Standard Errors (SE) for Large and Small Samples*

	ρ :							
Test Composition	0.1		0.2		0.3		0.4	
[3 : 5; 5; 5]	#	SE	#	SE	#	SE	#	SE
	Complete linkage							
N=200	9	.053	9	.100	3	.099	0	-
N=100,000	10	.003	10	.004	0	-	0	-
	Average linkage							
N=200	1	.082	0	-	0	-	0	-
N=100,000	1	.000	1	.000	0	-	0	-
	Within-groups linkage							
N=200	3	.037	4	.094	0	-	0	-
N=100,000	10	.000	10	.005	0	-	0	-
	Scale linkage							
N=200	9	.039	9	.084	3	.081	1	-
N=100,000	10	.000	10	.007	10	.005	0	-

Note. '#' represents number (out of ten) times the correct clusters were found using pre-defined stopping rules. 'SE' represents the standard error of the statistics found for the correct solutions. '-' denotes SE cannot be determined.

Table 2.5: *Ranges of c Yielding the Correct Dimensionality Using Four Hierarchical Clustering Methods for Simulated Data Sets*

	ρ :				
Test Composition	0.1	0.4	0.7	0.2/0.6	0.4/0.6
	Complete Linkage				
[3 : 5; 5; 5]	.04–.41	.14–.41	.25–.42	.21–.41	.21–.42
[3 : 5; 5; 10]	.05–.41	.17–.41	.31–.42	.25–.42	.26–.41
[3 : 10; 10; 10]	.03–.41	.12–.41	.22–.43	.18–.42	.19–.42
	Average Linkage				
[3 : 5; 5; 5]	.08–.28	.17–.28	.24–.28	.21–.28	.20–.28
[3 : 5; 5; 10]	.08–.28	.18–.28	.24–.28	.22–.28	.19–.27
[3 : 10; 10; 10]	.08–.27	.18–.28	.26–.28	.23–.27	.22–.28
	Within-groups Linkage				
[3 : 5; 5; 5]	.36–.62	.45–.63	.54–.64	.51–.62	.43–.64
[3 : 5; 5; 10]	.41–.63	.51–.63	.59–.64	.55–.63	.56–.64
[3 : 10; 10; 10]	.39–.67	.49–.68	.59–.69	.55–.67	.55–.67
	Scale Linkage				
[3 : 5; 5; 5]	.32–.60	.42–.61	.51–.62	.47–.60	.48-.61
[3 : 5; 5; 10]	.38–.61	.48–.62	.56–.62	.52–.61	.53–.61
[3 : 10; 10; 10]	.36–.64	.46–.65	.56–.66	.52–.64	.52–.65

As indicated by the preceding results, the use of the hypothesized stopping rules may not be the desirable approach to determine the number of dominant dimensions of a data set. The best moment to stop the clustering process depends on several characteristics of the data: that is, on the properties of the items (e.g., discrimination) and of the subjects (e.g., correlations between their traits, sample size). Explorative approaches that analyze the process of clustering may be more appropriate than confirmatory approaches that use predefined stopping rules in determining the moment to stop the clustering.

In the third type of results scree plots of the proximities at each hierarchical step are presented. Figure 2.1 depicts the scree plot of the number of clusters and maximum proximity using complete linkage, average linkage, within-groups linkage and scale linkage on the simulated item response data having unequal numbers of items per latent trait and five different ρ conditions. Each plot depicts the drop in proximity (i.e., $\max(H_{O_v O_w})$ when two objects are joined into one cluster at each cluster step[3]. We expect the proximity function to drop off when objects are joined because H generally decreases when more items are added to a scale. However, when objects sensitive to different traits are joined a larger drop is expected than when objects sensitive to the same latent trait are joined. Thus, with the sharp-drop off criterion we try to capture the drop in proximity that cannot be explained by the reduced correlation due to the numbers of items in the set(s), but by multidimensionality. One should note, however, that a sharp drop in proximity may not only occur for reasons of multidimensionality, but also, for example, when there are one or two items having low discrimination in a set having high discrimination.

For equal numbers of items per latent trait, the complete linkage, within-groups linkage and scale linkage plots indicate that the true dimensionality was found if the solution before a sharp drop in $\max(H_{O_v O_w})$ was used (at 3 clusters). The average linkage pattern was more difficult to interpret. Several larger drops can be seen, for example, for $\rho = 0.4$ and 0.7 a sharp drop in $\max(H_{O_v O_w})$ can be observed at the 6-cluster solution; and a less obvious drop can be observed for $\rho = 0.1$ at the three-cluster solution. Thus, for this method it is more difficult to decide when to stop clustering.

The plots for equal numbers of items were very systematic in the sense that comparable objects were joined for each dimension. One may see in Figure 2.1 that $\max[H(_{O_v O_w})]$ does not change much for the first three steps, for example, because

[3]Note that Figures 2.1 and 2.2 do not depict the drop in proximity when one item is added to a single scale, but the drop when objects are joined. One or both of these objects may contain a single item.

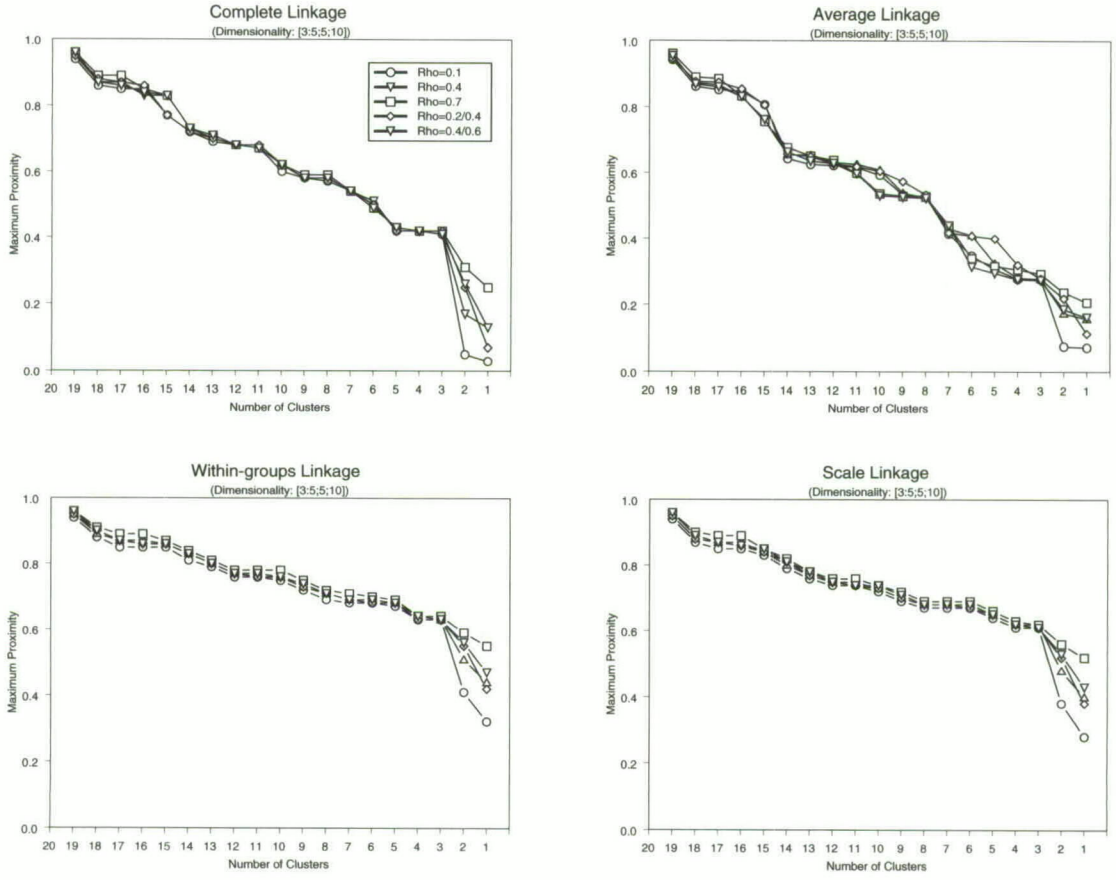


Figure 2.1: *Scree Plots of Number of Clusters and Maximum Proximity for the four hierarchical approaches.*

for each dimension items with the same item characteristics (i.e., discrimination and difficulty) were joined. Because for unequal numbers of items per trait (i.e., true dimensionality: [3:5;5;10]) the items characteristics were not the same, the patterns were less systematic (see, Figure 2.1). Although the pattern was less systematic, the general results for unequal numbers of items per latent traits remain the same as for equal numbers.

2.6 Empirical Example

The five clustering methods were compared using an empirical example. We used a dichotomized 15-item subset of the International Survey on Adult Crying (ISAC-A; Becht, Poortinga, & Vingerhoets, 2001) obtained for $N = 3,896$ subjects from 30 countries. The questionnaire consists of 54 items about common events and feelings that may induce crying. For clarity of presentation this small subset of items was used. We ignored any cultural diversity in our analysis and missing data were deleted listwise from the analysis.

In an empirical study, the theoretical constructs on which the test or questionnaire is based can be used to form hypotheses about the true dimensionality. From previous studies on the ISAC-A, two (Becht et al., 2001; Scheirs & Sijtsma, 2001) or three (Scheirs & Sijtsma, 2001) types of items can be distinguished: *distress*, representing emotions or situations which are unpleasant, for example, "I cry when having been humiliated or insulted" (item 24); *sadness*, representing emotions or situations which are sad, for example, "I cry when I feel sad" (item 1); and *joy*, representing emotions or situations which are happy, for example, "I cry when a movie or a television program has a happy ending" (item 13). Becht et al. (2001) did not differentiate between the distress and sadness items. The H coefficient of these subtests were $H_{distress} = 0.50$ (8 items), $H_{sadness} = 0.47$ (3 items), and $H_{joy} = 0.34$ (4 items). The sum scores of the responses on the distress and sadness subsets were moderately related: their Pearson product moment correlation was $\rho = 0.622$. The correlations between the responses on the distress and joy subtests (i.e., $\rho = 0.393$), and the sadness and joy subtests (i.e., $\rho = 0.357$) were much weaker.

We analyzed the crying data using the five clustering procedures. For sequential clustering the same conditions were used as in the simulation study (i.e. $c = 0.3$, $\alpha = 0.05$). For hierarchical clustering, new stopping rules were specified using the ranges in Table 2.5; these are, $c^{complete} = 0.35$, $c^{average} = 0.25$, $c^{within} = 0.60$, and $c^{scale} = 0.55$. These new stopping rules are based on the results of our simulation study which may not be representative for empirical data.

Also information about the process of clustering was obtained by making use of graphical analysis.

2.6.1 Results

Table 2.6 shows the results of the dimensionality analysis of the ISAC-A data. The first column shows the cluster solution, the second column shows the theoretical constructs of each item per cluster, and the third column shows the $\max(H_{O_v O_w})$ of the newly clustered objects. The letters 'D', 'J', and 'S' represent distress, joy or sadness items, respectively. As can be seen from Table 2.6, no clustering method found the theoretical dimensionality. Sequential clustering resulted into a 2-cluster solution, confirming that the responses on the distress and sadness items are substantially correlated. The hierarchical clustering methods yielded different results depending on the proximity and the stopping rule used for clustering. Using the new values for the stopping rules (denoted with an asterisk in Table 2.6), the hierarchical methods found one large scale containing most distress-items and one sadness-item, as well as several smaller scales. One sadness-item was always found in the cluster that mainly contained distress-items because that item had the highest H_{jk} with one of the distress items making this pair the starting set of all clustering methods. This shows a property of the items that is not detected when testing the homogeneity of the clusters confirmatory.

Scree plots of the ISAC-A data (see, Figure 2.2) and complete linkage, average linkage and within-groups linkage dendrograms (see, Figures 2.3[4]) may provide us with more reliable information about what solution to use. The dendrograms presented in Figure 2.3 depict the process by which items were joined, as well as the change in proximity between steps. The actual distances were rescaled to numbers between 0 and 0.25, preserving the ratio of differences between steps (SPSS, 1998).

Figure 2.2 shows complete linkage had the most informative plot for determining the number of underlying traits: a clear cut-off point can be found at the three-cluster solution (also see, Figure 2.3). Going from four to three clusters did not result into a large decrease in $\max(H_{O_v O_w})$ and, therefore, the three-cluster solution was the dimensionality according to complete linkage. Within-groups linkage and scale linkage showed only minor drops in $\max(H_{O_v O_w})$ at $K = 3$ and were therefore less clear with respect to the dimensionality of these items. The pattern of average linkage was difficult to interpret: there were several small

[4]We have not included a scale linkage dendrogram because this cannot be replicated using SPSS and thus is not readily available for applied researchers.

Table 2.6: *Number of Clusters, Theoretical Basis of Items and Maximum Proximity Using Five Clustering Methods for ISAC-A Data*

Clust.	Theory	Prox.
Sequential clustering (c=0.3)		
2	[J2,J3,J4] [D6,D7,D8,D10,D11,D12,D13,D14,J15,S1,S5,S9]	-
Complete linkage		
4*	[J3,J4][D14,D15,S5] [D7,D8,D10,D11,D13,D6D12,S9] [S1]	0.37
3	[J2,J3,J4][D14,J15,S1,S5] [D7,D8,D10,D11,D13,D6,D12,S9]	0.32
2♯	[J2,J3,J4] [D6,D7,D8,D10,D11,D12,D13,D14,J15,S1,S5,S9]	0.19
Average linkage		
4*	[J3,J4] [D7,D8,D10,D6,D12,J15,J2,S1,S9] [D14,S5] [D11,D13]	0.28
3	[J3,J4][D14,S5][D11,D13,D7,D8,D10,D6,D12,J15,J2,S1,S9]	0.24
2	[J3,J4,D14,S5][D11,D13,D7,D8,D10,D6,D12,J15,J2,S1,S9]	0.23
Within-groups linkage		
10*	[J3][J4][D6,D7][D8,D10,D13,D12,S9][D11][J15][D12][J2][S1][S5]	0.61
...	...	...
4	[J3,J4][J15][D6,D7,D8,D10,D11,D12,D13,D14,S1,S5,S9][J2]	0.49
3♯	[J3,J4][J15,D6,D7,D8,D10,D11,D12,D13,D14,S1,S5,S9][J2]	0.48
2	[J3,J4][J2,J15,D6,D7,D8,D10,D11,D12,D13,D14,S1,S5,S9]	0.45
Scale linkage		
8*	[J2] [J3] [D14] [J4] [J15] [S1] [S5] [D7,D8,D10,D11,D13,D6,D12,S9]	0.56
...	...	...
4	[J3,J4][D6,D7,D8,D10,D11,D12,D13,J15,S1,S5,S9][D14][J2]	0.49
3♯	[J3,J4][D6,D7,D8,D10,D11,D12,D13,D14,J15,S1,S5,S9][J2]	0.47
2	[J3,J4][D6,D7,D8,D10,D11,D12,D13,D14,J15,J2,S1,S5,S9]	0.44

Note. *=result for hypothesized values of the stopping rules. ♯= result for general stopping rule, $H_j < 0.3$. The solutions still include missfitting items.

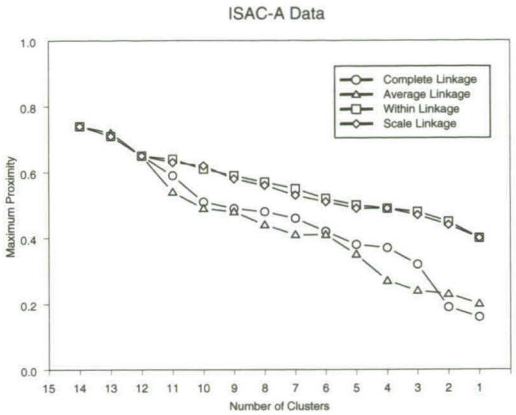


Figure 2.2: *Scree Plots of Number of Clusters and Maximum Proximity Using Complete Linkage, Average Linkage, Within-groups Linkage and Scale Linkage of ISAC-A Data.*

decreases, but no clear cut-off point.

The relationship between the scores operationalized by the methods and correlations between the sets can help to interpret the results of the scree plots. Complete linkage joins the two objects having the highest minimum H_{jk} at each clustering step; thus, clustering is based on the relationship between each pair of variables. The other methods use proxies for H_j or H that, in general, represent the relationship between more than two variables. Thus, the other methods may average out the differences that complete linkage focusses on. Complete linkage could therefore depict a sharp drop-off where the other methods could not.

The lack of a clear cut-off point for all methods except complete linkage can be explained by the relatively high correlations between the sets. In particular, the sets measured common variables and, therefore, the proximity did not drop much when items sensitive to these variables were clustered. Although a sharp drop-off was observed for complete linkage at $K = 3$, the method's plot also indicates that the sets were moderately correlated. The proximities' values at $K = 2$ and $K = 1$, which were still quite high (probably even significantly positive for this sample size; see Condition 1), indicated this.

Figure 2.3 illustrates that the process of clustering is different in the various methods. For example, sequential clustering, complete linkage, and scale linkage (not shown) identify joy as a separate cluster in an early stage of the clustering process, whereas average linkage and within-groups linkage do not. To summarize

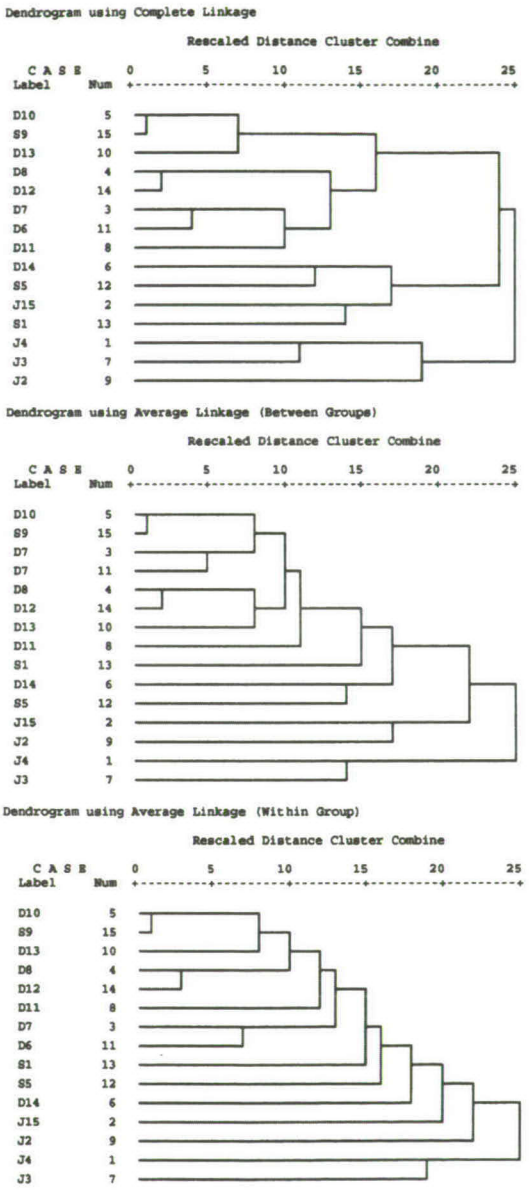


Figure 2.3: *Dendrogram of Complete Linkage, Average Linkage and Within-groups Linkage of ISAC-A Data.*

the results of Figure 2.3, complete linkage was most useful as a method for determining the number of underlying variables, and complete linkage as well as scale linkage methods were successful for correctly identifying unidimensional sets.

When looking at the $K = 2$ solutions in Table 2.6, one can see that complete linkage leads to the same result as sequential clustering. One may note that the $K = 2$ solution of complete linkage combines two clusters of the $K = 3$ solution (i.e., the best solution according to the scree plot). Even though the results for $K = 2$ were the same, complete linkage should be preferred to sequential clustering because with complete linkage more certain statements can be made about the dimensionality of the items. In sequential clustering, it is not known whether the two clusters were combined (i.e., into the largest cluster) because this yields the strongest Mokken scale or because of the sequential nature of the item selection procedure (i.e., forming clusters one at the time). In hierarchical clustering this information is known because $H_{O_v O_w}$ for all combinations of objects are calculated and only the two objects that maximized $H_{O_v O_w}$ are combined.

Using complete linkage and interpreting its graphically depicted result, the three-cluster solution should be considered to be the best (e.g., clear results in scree plot, satisfies Mokken scale conditions). Based on substantive grounds or on the correlations between the sets, however, one might prefer the two-cluster solution with the lower within-cluster homogeneity.

2.7 Discussion

In this study, it is investigated whether hierarchical clustering improves sequential clustering that is the standard in Mokken scale analysis. The simulation study showed that all four HCA-methods (i.e. complete linkage, average linkage, within-groups linkage and scale linkage) were able to find the true dimensionality. However, the success of the hierarchical methods depended on the stopping rules of the clustering process. Scale linkage was the promising hierarchical clustering method because it found the dominant dimensionality for most data. Complete linkage also was promising because for this method a large range of $c^{complete}$ values lead to the true dimensionality, and because scree plots were interpretable for this method. Finally, the general stopping rule H_j used in combination with each of the HCA-methods seems to work well. This was to be expected because that rule is most closely related to the Mokken scaling conditions. In the empirical study, not all methods yielded the same results. Here, the underlying traits seemed to be substantially correlated and, again, only complete linkage displayed an interpretable scree plot.

There are two reasons why scale linkage and complete linkage are an improvement over sequential clustering. First, these methods yield better results, meaning that items sensitive to the same latent trait are more often collected in one cluster using these methods than using sequential clustering. Second, the clustering process is more informative: it shows which objects are joint at what step; and it shows the relative difference in maximum proximity between clustering steps. Using graphical methods (i.e., scree plots or dendrograms) the 'best' dimensionality for a particular set of items can be found; that is, the solution before a sharp drop in maximum proximity is seen. One should be aware, however, that the best solution in terms of relative difference may not satisfy the Mokken scaling conditions. Additionally, the presence of a sharp drop may depend on characteristics of the data. For example, for highly correlated latent traits no sharp drop can be expected. Thus, the decision about which dimensionality to use should also be based on H_j and H values of clusters. The process information of sequential clustering is less informative. It does not inform us whether any sub-clusters (i.e., these are the objects contained in a cluster) exist. Using hierarchical clustering methods one has information about the sub-clusters and their scalability, and this information can be used to find the true dimensionality.

We ignored sample fluctuation issues for the largest part of this chapter by using very large sample sizes. For small sample sizes (say, $N = 200$) it may occur that the H_{jk} for some low discrimination items are negative due to sample fluctuation rather than because they are sensitive to different latent traits. The same line of reasoning goes for values of H_j near c. For a few cells of the simulation study, the stability of the results for $N = 200$ was investigated and this confirmed what was expectated. In future research it may be worthwhile to address the impact of sample fluctuation on dimensionality assessment more explicitly.

As is common in sequential item selection, when creating scales it is wise to make use of other available information. More specifically, here one can use information about the clustering process as presented in a dendrogram, substantive information, and methods that may be used to search for specific violations of MH model can provide additional information about the dominant underlying dimensionality of data.

Rather than a hierarchical procedure a non-hierarchical procedure could have been used where an overall criterion is optimized. For example, one could calculate H_j for all items j and a given number of k clusters, and assign item j to that cluster that maximizes H_j. The advantage of such a procedure is that not only objects but also items within objects are compared. In future research non-hierarchical procedures are addressed.

Chapter 3

Some Alternative Clustering Methods for Mokken Scale Analysis

Abstract

In this chapter three methods for finding the dimensionality of a data matrix based on different types of algorithms (sequential, hierarchical, and non-hierarchical) were discussed. For each method different measures were suggested to find one or more sets of dichotomous items that satisfy the conditions of a Mokken scale. It was illustrated that non-hierarchical clustering resolves some problems associated with sequential and hierarchical clustering in yielding an optimal solution and in finding the true dimensionality.

This chapter is based on: Van Abswoude, A.A.H. & Vermunt, J.K. (2003). Some Alternative Clustering Methods for Mokken Scale Analysis. In H. Yanai, A. Okada, K. Shigemasu, Y. Kado, & J.J. Meulman (Eds.) *New developments in Psychometrics* (pp. 625-630). Tokyo: Springer.

3.1 Introduction

In measurement and scaling it is important to measure one single latent trait at one time. Otherwise, unless the exact relationship between the latent variables can be modelled, it will be difficult to assign meaningful scores to subjects. In practice, however, researchers are often confronted with data matrices sensitive to multiple latent traits. For example, a test that measures *crying* may contain items on sub-traits such as *distress*, *sadness*, and *joy*.

Mokken scale analysis (MSA; Mokken, 1971) may be used to find sets of items that form a single unidimensional scale. The MSA-software, MSP (Molenaar & Sijtsma, 2000), uses a sequential clustering algorithm to find sets of items (clusters) that satisfy the conditions of a Mokken scale. A drawback of this algorithm is, however, that it may not yield the *optimal solution*; that is, the solution that maximizes a certain objective function. If the objective function is correct, the optimal solution should reflect the true dimensionality of the data matrix. A practical implication of using the sequential algorithm is that the user may obtain a set of items that measures more than one trait.

In this chapter, two alternative clustering methods for MSA are discussed, hierarchical and non-hierarchical clustering, which may do a better job in finding the optimal solution. In the following sections, it is described how the clustering methods work and discuss how the MSA conditions can be imposed within each of these methods so that solutions may reflect the true dimensionality and satisfy the MSA conditions. In addition, the ability of the methods in finding the optimal solution will be illustrated by means of three small simulated examples.

3.2 Nonparametric IRT Framework

In nonparametric item response theory (IRT) it is assumed that a single underlying latent trait (θ) governs the responses on a set of items (unidimensionality, UD). Further it is assumed that given any value of θ the responses of an individual on a set of items are statistically independent (local independence, LI). Lastly, it is assumed that there is a monotonely nondecreasing relationship between the probability of answering an item correctly and θ (monotonicity, M). A set of items that satisfy UD, LI and M are denoted as monotone homogeneous (MH; Mokken, 1971). The monotone homogeneity model allows the ordinal measurement of θ by means of the total test scores of individuals (Grayson, 1988). The total test score is defined by $X_+ = \sum X_i$, with X_i denoting an individual's score on item i ($i = 1, ..., I$). Within the nonparametric IRT framework, the response probability

given θ, known as the item response function (IRF), does not need to have a particular shape such as the logistic as long as the items are MH.

Mokken scale analysis, which is a nonparametric IRT method for scale analysis, uses Loevinger's coefficient of homogeneity to quantify the strength of the association within the responses to a pair of items. Let items i and j be two binary items having item scores 0 or 1. Let π_i represent the probability of answering item i correctly, and π_{ij} the probability of answering both item i and j correctly. Items are ordered such that $\pi_i \leq \pi_j$, for all $i < j$. Let $\pi_{ij}^{(0)} = \pi_i(1 - \pi_j)$ if $\pi_i \leq \pi_j$ [and $\pi_{ij}^{(0)} = (1 - \pi_i)\pi_j$ if $\pi_i > \pi_j$]. The pairwise scalability coefficient H_{ij} for items i and j is defined as

$$H_{ij} = \frac{\pi_{ij} - \pi_i \pi_j}{\pi_i(1 - \pi_j)}. \tag{3.1}$$

This equals the covariance between items i and j divided by the maximum covariance given the marginal score distributions of items i and j (Mokken, 1971; Molenaar and Sijtsma, 2000). The H_i for item i can be written as

$$H_i = \frac{\sum_{j \neq i}(\pi_{ij} - \pi_i \pi_j)}{\sum_{j \neq i} \pi_{ij}^{(0)}} = \frac{\sum_{j \neq i} \pi_{ij}^{(0)} H_{ij}}{\sum_{j \neq i} \pi_{ij}^{(0)}} \tag{3.2}$$

as in Mokken (1971, p. 150). The scale H is defined as

$$H = \frac{\sum_i \sum_{j \neq i}(\pi_{ij} - \pi_i \pi_j)}{\sum_i \sum_{j \neq i} \pi_{ij}^{(0)}} = \frac{\sum_i \sum_{j \neq i} \pi_{ij}^{(0)} H_{ij}}{\sum_i \sum_{j \neq i} \pi_{ij}^{(0)}}. \tag{3.3}$$

For more information about the theoretical basis and the sampling distribution of the H coefficient the reader is referred to Mokken (1971) or Molenaar and Sijtsma (2000).

A set of I items is called a *Mokken Scale* (Mokken, 1971, p. 184) if all items satisfy the following two conditions,

Condition 1 $\text{cov}(X_i, X_j) > 0$, for all $i \neq j$, and

Condition 2 $H_i \geq c$, for all i, where c is a user-defined constant between 0 and 1.

From the MH model follows that $\text{cov}(X_i, X_j) \geq 0$ (Holland and Rosenbaum, 1986), but the converse does not hold. The second condition serves the practical purpose of only including items into a scale that sufficiently discriminate. In practice, $c = 0.3$ is usually sufficiently high to measure a single latent trait.

Below, three types of clustering procedures are presented (i.e., sequential clustering, hierarchical clustering, and non-hierarchical clustering) for finding the dimensionality of a data matrix using the H coefficient as a measure of association. Each method uses H_{ij}, π_i, and π_j as input data; and each must satisfy the two conditions of a Mokken scale. In the next sections, it is described how these conditions can be imposed, as well as what the limitations of these methods are when searching for the dimensionality of a set of items.

3.3 Sequential Approach

Sequential clustering takes the following stepwise course. In the first step (Step 1), the two items that form the start set for the first scale are selected. This is the item pair with the highest significantly positive H_{ij} (Equation 3.1) that satisfies Conditions 1 and 2. In the next step (Step 2), items are added one at the time to this start set. More precisely, this is the item that yields the highest H with the previously selected items (Equation 3.2) that satisfies Conditions 1 and 2. This process of adding items that yields the highest H continues until no item remains. When this happens the first scale has been formed. As long as scalable items remain, subsequent scales may be formed by repeating Steps 1 and 2 using the remaining items. The procedure terminates when no scalable items are left.

The sequential procedure works quite well when one is interested in finding only one single Mokken scale, for example when a data matrix measures one dominant latent trait, and possibly one or more nuisance latent traits. When searching for multiple Mokken scales in a multi-trait context, however, the sequential nature of the procedure may yield suboptimal solutions: the solution that overall yields the highest H_i is not obtained. The reason that suboptimal solutions may be obtained is that sequential clustering forms the clusters one at the time. As a result, some items may be collected in Cluster 1 (clustering continues until scaling criteria are no longer satisfied), although H_i may have been higher when joined with items in Cluster 2.

3.4 Hierarchical Approach

Agglomerative hierarchical clustering analysis (HCA; e.g., Everitt et al., 2001) seems to be a useful alternative because it can yield multiple clusters simultaneously, where sequential clustering could not. Starting point of a HCA is a data matrix containing proximities between items i and j. The proximities in our case

are based on the H coefficient and will be discussed in more detail later on. At each hierarchical step, the two clusters with the highest proximity are joined. This means that at any hierarchical step two single items may be joined to form one new cluster, a single item may be joined with an existing cluster of items, or two clusters may be joined into a single larger cluster. This process continues until some previously defined criterion is met or until all items are in one single cluster.

In the following methods, four types of proximities may be used to form Mokken scales. The first three methods can be reproduced using the H_{ij}-matrix in combination with standard clustering procedures of most statistical packages, including SPSS, SAS, and S-plus. For the fourth method dedicated software was written in PASCAL. Before the proximities are presented, however, first some notation is needed. Let k and l represent two clusters, let I_k represent the number of items in cluster k, I_l represent the number of items in cluster l, and let H_{kl} represent the proximity between clusters k and l.

In *complete linkage* the proximity between clusters k and l is defined as,

$$H_{kl}^{complete} = \min(H_{ij}), \text{ where } i \in k \text{ and } j \in l.$$

This method joins the two clusters for which the lowest H_{ij} of the two clusters is maximized. This definition of proximity is intuitively attractive because it produces scales for which the $\min(H_{ij})$ satisfies some minimal requirement.

Average linkage defines the proximity between clusters as

$$H_{kl}^{average} = \frac{\sum_{i \in k} \sum_{j \in l} H_{ij}.}{I_k I_l}.$$

As can be seen, $H_{kl}^{average}$ is the unweighted average of the bivariate H_{ij} *between* the items in cluster k and the items in cluster l. This measure can be viewed as a proxy for the average H_i in a cluster.

Within-groups linkage defines the proximity of two clusters k and l as the unweighted average of the H_{ij} of all items *within* k and l; that is,

$$H_{kl}^{within} = \frac{\sum_i \sum_{j \neq i} H_{ij}}{(I_k + I_l)(I_k + I_l - 1)}, \text{where } \{i, j\} \in k \cup l.$$

This proximity can be seen as a proxy for H as defined in Equation 4.5.

The fourth method, *scale linkage* uses the scale H (Equation 4.5) of the possible new cluster that is obtained by joining two clusters as proximity measure. Written in terms of clusters k and l the proximity in scale linkage is defined as,

$$H_{kl}^{scale} = \frac{\sum_i \sum_{j \neq i} \pi_{ij}^{(0)} H_{ij}}{\sum_i \sum_{j \neq i} \pi_{ij}^{(0)}},$$

where $\{i, j\} \in k \cup l$, and for all $i > j$. Different from sequential clustering is that, unless a stopping rule is used, the HCA will continue clustering until all items are joined into one large cluster. For instance, the Mokken scaling conditions (especially, Condition 2) can be used to terminate the clustering process. In that case, clustering stops when the conditions are no longer satisfied. Alternative methods to decide when to stop the HCA can also be proposed, but are beyond the scope of this chapter. In this chapter the Mokken scale conditions were used as stopping rule.

Unfortunately, hierarchical clustering may also yield suboptimal solutions because clusters that have been formed in previous steps remain intact in subsequent steps. More precisely, a set of items that was clustered at an earlier stage may not be homogeneous with respect to items that were added later.

3.5 Non-Hierarchical Approach

Non-hierarchical clustering refers to a class of algorithms where multiple clusters are formed simultaneously and single units within an object (i.e., items) can be moved from one cluster to another. The method uses a criterion, which is based on the H coefficient, to evaluate the quality of a partition $\mathcal{P}_t$ at iteration t.

Let $\delta_{ik}(\mathcal{P}_t) = 1$ if $i \in k$ (where $k = 1, ..., K$), and $\delta_{ik}(\mathcal{P}_t) = 0$ if $i \notin k$ at $\mathcal{P}_t$. In addition, let $H_{i|k}$ be matrix reflecting the conditional homogeneity of each item i with respect to the items in each cluster k. A criterion for evaluating the quality of a K-cluster at partition $\mathcal{P}_t$ may, for instance, be

$$\text{Criterion1}(\mathcal{P}_t) = I^{-1} \sum_{i=1}^{I} \sum_{k=1}^{K} \delta_{ik}(\mathcal{P}_t) H_{i|k}. \tag{3.4}$$

The goal of the non-hierarchical clustering procedure is to search for that partition that maximizes Criterion1; that is, one intends to join each item into the cluster such that the highest H_i is obtained for all items.

In this chapter, a k-means type algorithm was used to assign items to clusters. This clustering method begins with an initial configuration ($t = 0$) in which I items are randomly assigned to K clusters, and the quality of $\mathcal{P}_0$ is evaluated using Criterion1($\mathcal{P}_0$). In each iteration step, one item i may be moved to another cluster k and Criterion1($\mathcal{P}_1$) is evaluated. Different rules may be used to move an item i to another cluster k. For example, one could move the item to that cluster for which the improvement in $H_{i|k}$ is the best. In the subsequent iterations this evaluating and maximizing of Criterion1($\mathcal{P}_t$) is continued until the criterion can no longer be improved.

This procedure can be further refined by adding a random component to the process of assigning items to clusters, thereby reducing the probability of ending up in a local maximum: a stochastic process could be used for the assignment of items to clusters. This means that in the first iterations, Criterion1 may deteriorate from one iteration to the next (e.g., an item is moved to a cluster for which $H_{i|k}$ is not the highest). For later iterations improvements of Criterion1 are more likely. This random element in the composition of clusters is important because it may yield combinations of items that otherwise would not have been found.

Other criteria could also be used in this context. The following criterion was formulated for finding the partition having the highest H_{ij} within clusters. We adapted Kim's statistic for this purpose, which is aimed at finding clusters of items that are locally independent (Kim, 1994). Let $\delta_{ijk}(\mathcal{P}_t) = 1$ when items i and j are joined into cluster k at $\mathcal{P}_t$, and -1 otherwise. Then, Criterion 2($\mathcal{P}_t$) is defined as

$$\text{Criterion2}(\mathcal{P}_t) = \frac{2}{I(I-1)} \sum_{1 \leq i < j \leq I} \delta_{ijk}(\mathcal{P}_t) H_{ij}.$$

Criterion2 can also be maximized using the k-means type procedure which was presented before. In the simulation study, however, only Criterion1 was used (i.e., without the proposed refinements).

3.6 Simulation Study

The performance of the three general procedures (i.e., sequential, hierarchical, and non-hierarchical clustering) using their specific definitions of dimensionality is shown for three small generated tests. We used a multidimensional extension of the two-parameter logistic item response theory model (M2-PLM; e.g., Reckase, 1985) to generate 1,000 item responses on sets of six, ten and twenty items. Even though the data were generated, the used item parameter values are representative for true test data.

Test 1 consists of 6 items (Items1, Item2, $\cdots$, Item6) and two latent traits, θ_1 and θ_2. Item1 and Item2 are strongly related to θ_1, Item3 is weakly related to θ_1 and strongly related to θ_2, and Item4, Item5 and Item6 are moderately related to θ_2 but not related to θ_1.

Test 2 consists of 10 items, and three latent traits, $\theta_1, \cdots, \theta_3$. Item1, $\cdots$, Item5 are moderately related to θ_1, Item5 and Item6 are moderately related to θ_2, and Item6, $\cdots$, Item10 are moderately related to θ_3. The second latent trait,

θ_2, has the function of a nuisance trait: that is, it is included in order to make the detection of the items measuring θ_1 and θ_3 more difficult.

Test 3 consists of 20 items, and two latent traits θ_1 and θ_2. Item1, $\cdots$, Item10 are strongly related to θ_1, Item11, $\cdots$, Item20 are strongly related to θ_2 and weakly related to θ_1.

The latent traits in the presented tests are assumed to be uncorrelated. We evaluated the performance of the methods as to whether the solutions of sequential, hierarchical and non-hierarchical clustering correspond with the simulated dimensionality. Notation $[K : I_1; I_2; ...; I_K]$ is used to reflect the structure of an test, where K equals the number of clusters and I_k $(k = 1, ..., K)$ equals the number of items in each cluster. The preferred solution of Test 1 using $c = 0.3$ in Criterion2 is $[2 : 2; 4]$; that is, Item1 and Item2 are in Cluster 1 and Item3, $\cdots$, Item6 in Cluster 2. We look at two preferred solutions for Test 2: $[2 : 5; 5]$ using $c = 0.2$ in Condition 2 (i.e., Item1, $\cdots$, Item5 and Item6, $\cdots$, Item10; and $[3 : 4; 2; 4]$ using $c = 0.3$ in Condition 2 (i.e., Item1, $\cdots$, Item4, Item5, Item6, and Item7, $\cdots$, Item10). The different values of c reflect two possible ways of defining a Mokken scale: one is moderately strict ($c = 0.3$) and one is less strict ($c = 0.2$). The preferred solution for Test 3 using $c = 0.3$ is [2:10;10].

The default settings for MSP (Molenaar & Sijtsma, 2000) and HCA was used. For the k-means method the version portrayed in combination with Criterion1 was used.

3.7 Results

Table 3.1 shows the simulated dimensionality solution (i.e., highest H for all clusters) and the clustering solutions obtained with sequential clustering, hierarchical clustering, and non-hierarchical clustering. The first column gives the method used for clustering, the other columns give the results for Test 1, 2 and 3.

For Test 1, all methods, except sequential clustering, yielded the predefined preferred solution. The reason that the first cluster, obtained with sequential clustering, contained one extra item was that this item still satisfied the scaling criteria for the first cluster, although it measures θ_2. Forming multiple clusters simultaneously (i.e., hierarchical and non-hierarchical clustering) was sufficient to find the preferred solution.

In Test 2, both the $K = 2$ and the $K = 3$ results are presented for each method. Sequential clustering yielded the simulated dimensionality for both $K = 2$ and $K = 3$. With the hierarchical procedures either the $K = 2$ or the $K = 3$ solution corresponded with the optimal solution. One may note that with a hierarchical

Table 3.1: *Number of Clusters and Number of Items per Cluster Using Sequential Clustering, Four Hierarchical Clustering Methods and Non-Hierarchical Clustering for Three Tests*

Method	Test 1	Test 2	Test 3
Preferred solution	[2: 2;4]	[2: 5;5], [3: 4;2;4]	[2: 10;10]
Sequential	[2: 3;3]	[2: 5;5], [3: 4;2;4]	[2: 6;14]
Complete linkage	[2: 2;4]	[2: 6;4], [3: 4;2;4]	[2: 10;10]
Average linkage	[2: 2;4]	[2: 6;4], [3: 4;2;4]	[2: 10;10]
Within-groups linkage	[2: 2;4]	[2: 5;5], [3: 1;4;5]	[2: 10;10]
Scale linkage	[2: 2;4]	[2: 5;5], [3: 1;4;5]	[2: 10;10]
Simple + Criterion1	[2: 2;4]	[2: 5;5], [3: 4;2;4]	[2: 10;10]

procedure it is not possible that both the $K = 2$ and the $K = 3$ solution are optimal because the $K = 2$ solution is obtained by combining the two clusters of the $K = 3$ solution. The non-hierarchical method yielded two optimal solutions because clusters are formed simultaneously and individual items are assigned to the clusters they fit most.

In Table 3.1 one can see that the results of Test 3 are similar to Test 1, except that here an entire subset of four items (in stead of one single item) was incorrectly classified when using the sequential method. The results of Test 3 illustrate that the mechanisms that were responsible for the results of Test 1 and 2 may also be active in somewhat larger data matrices. Obviously, the examples can be easily be extended to even more items.

3.8 Conclusion

Three types of clustering methods for finding the dimensionality of a set of items were presented in this chapter. Each method was adapted to yield clusters that satisfy the Mokken scale conditions. As illustrated, non-hierarchical clustering resolves the problems associated with sequential and hierarchical clustering.

The non-hierarchical clustering algorithm used in the simulation study may yield local maxima. Introducing randomness in the assignment of items to clusters may be the remedy for this problem that deserves further study.

Chapter 4

Assessing Dimensionality by Maximizing H Coefficient Based Objective Functions

Abstract

The program MSP may not always reflect the underlying dimensionality of data. One or more features of MSP - the H coefficient, the Mokken scale conditions and the algorithm - may explain this result. In this chapter three new H-based objective functions that use slight reformulations of Mokken scale analysis in the unidimensional and multidimensional case were introduced. Deterministic and stochastic non-hierarchical clustering algorithms were used to reduce the probability of obtaining suboptimal solutions. The scale conditions were dropped. A simulation study was conducted to investigate whether these methods can be used to determine the dimensionality structure of different types of data that vary with respect to item discrimination, item difficulty, the number of items per trait, and numbers of observations per test. Further, it was investigated whether deterministic and stochastic algorithms can yield global optimal solutions. The method that used the average within-scale H_i combined with a stochastic non-hierarchical clustering algorithm was the most successful in dimensionality assessment.

4.1 Introduction

Four nonparametric item response theory (IRT) methods can be used for dimensionality assessment of data with an ordinal measurement level. These are (in alphabetical order): DETECT (Kim, 1994; Zhang & Stout, 1999b), DIMTEST (Nandakumar & Stout, 1993; Stout et al., 2001), HCA/ CCPROX (Roussos et al., 1998) and MSP (Mokken, 1971; Molenaar & Sijtsma, 2000; Sijtsma & Molenaar, 2002). These methods have in common that they all use observable consequences of the monotone homogeneity model (MHM; Mokken, 1971). A set of items that satisfies the MHM is unidimensional (i.e., sensitive to a single latent trait), locally independent (i.e., the item responses are statistically independent given a fixed value of the latent trait), and meets the monotonicity assumption (i.e., the probability of answering an item correctly is an increasing function of the latent trait).

The methods vary in their focus on each of the particular MHM assumptions. A relaxation of nonparametric IRT's local independence assumption, denoted as weak LI (e.g., McDonald, 1985; Stout, 1987), is used to evaluate the relationship between item pairs in DETECT, DIMTEST and HCA/CCPROX. DETECT and HCA/CCPROX partition the items into clusters in such a way that locally independent sets of items are obtained as much as possible, and DIMTEST tests whether the responses fulfill weak LI. Although multidimensionality need not be the only possible explanation that weak LI does not hold, this approach may be the most direct approach available for assessing dimensionality. The methods have a few disadvantages; that is, these methods are not suitable for tests having few items and a modest sample size (sample sizes of 2,000 and less are regarded as being 'small'; Stout, 1987); they are sensitive to the strength of each scale (i.e., different number of items or different discrimination of the items between scales; see Van Abswoude et al., 2004); and their statistics have some bias (see Stout et al., 2001; Zhang et al., 2003; Roussos & Ozbek, 2003).

The focus of MSP is on creating scales rather than on dimensionality assessment. Items joined into a scale using MSP satisfy an observable consequence of the MHM on the one hand, and satisfy a user-defined condition on the other hand. The user-defined condition allows one to choose the minimal discrimination power of items in a scale. As a result, unlike the already discussed methods item selection is un-exhaustive and suitable for a large variety of test construction applications. The relationship between item scores is indexed by means of the H coefficient (Loevinger, 1948; Mokken, 1971, p. 148). This is a normed covariance which corrects for the maximum covariance that is possible given the marginal distribution of

the items. This coefficient need not be calculated for each latent trait value and thus requires fewer subjects and fewer items than the conditional statistics used in DETECT, DIMTEST and HCA/CCPROX. A disadvantage of the MSP method is that its sequential scaling algorithm may not yield the best possible solution. Technically, the best solution for one scale is the item set having the highest H value for as many items as possible and satisfying the scale conditions. Practically, this disadvantage could mean that the researcher may have obtained a scale having less strength to measure the underlying trait and/or consisting of fewer items than if an alternative algorithm were used. For a theoretical comparison of the four methods see Van Abswoude et al. (2004).

The main purpose of this chapter is to implement a new algorithm in MSP that allows us to keep the general focus of the method intact, but resolves the problems associated with the old algorithm. We use stochastic and deterministic versions of a non-hierarchical clustering algorithm (NHCA) for this purpose. New objective functions are introduced in which Mokken scale analysis is adapted to these new algorithms. These new objective functions define the problem of finding more than one scale (for short: multiple scaling) in a slightly different way than the sequential MSP method. The necessity of this redefinition and its consequences for scaling results are discussed later on.

Although the intend was to keep the scaling focus intact, in this chapter, the scale conditions are ignored most of the time. The main reason for this is simplicity: before restrictions are added to the problem, we first want to investigate how well the new functions work. A consequence of adopting this approach is that the new method is investigated as a dimensionality assessment tool. There are a number of advantages to this. If the scale conditions are not incorporated, weakly discriminating items, for which the assignment of items into scales is the most difficult, can be selected into scales, and thus the limitations of the method can be investigated. Further, using this approach we can find out whether the new function, the scale conditions, or the algorithm is responsible for splitting or joining of item pairs into clusters. Suggestions to extent the new Mokken scale method with the scale conditions is discussed in chapter 5.

Using a simulation study, it is determined how successful these methods are in finding the underlying dimensionality of a data matrix; this is the first research question. In particular, it is investigated the correspondence of the solution (i.e., the obtained sets of items or clusters) that maximized the functions, and the simulated dimensionality. The second research question is which algorithm can be used best. We judged the success of each algorithm by the number of times it yielded a global of near global solution, and by the number of iterations it needed.

4.2 Mokken Scale Analysis

Mokken scale analysis (MSA) uses Loevinger's H coefficient as a scalability coefficient (Loevinger, 1948; Mokken, 1971). The H coefficient can be understood when expressed in terms of Guttman errors (Guttman, 1950). Let $\boldsymbol{X} = (X_1, \cdots, X_I)$ be the vector of I binary scored item response variables (items), and let $\boldsymbol{x} = (x_1, \cdots, x_I)$ be their realizations (i.e., 0 denotes incorrect; 1 correct). In addition, let π_i denote the proportion subjects answering item i correctly and let all items be ordered such that $\pi_i \geq \pi_j$. Then, a subject answering an easy item i incorrectly and a difficult item j correctly produces a Guttman error. A larger number of Guttman errors than expected under the MHM in combination with the distribution of the latent trait(s) may be due to misfit of one or a few subjects (person-misfit; e.g., Meijer, 1994; Emons, 2003), the misfit of one or two items in specific subgroups of subjects (item bias), or the misfit of one or more items driven by unintended latent variables (multidimensionality; e.g., Stout, 1987). Person fit and item bias can be seen as special cases multidimensionality; that is, instead of an extra trait a grouping variable is introduced for one subject with respect to the total group, or for different subgroups.

Let N denote the number of subjects, F_{ij} the observed number of Guttman errors, and $E_{ij} = N\pi_i(1 - \pi_j)$ the expected number of Guttman errors under marginal independence. The H coefficient for an item pair (i, j) is defined as

$$H_{ij} = 1 - \frac{F_{ij}}{E_{ij}}. \tag{4.1}$$

One may note that $H_{ij} = 0$ when items i and j show exactly as many Guttman errors as expected under marginal independence; and $H_{ij} = 1$ when no Guttman errors are observed. The H coefficient can also be written as:

$$H_{ij} = \frac{\text{cov}(X_i, X_j)}{\text{cov}(X_i, X_j)\text{max}} \tag{4.2}$$

(Loevinger, 1948; Mokken, 1971). The H coefficient of a single item i in a scale consisting of I items equals

$$H_i = \frac{\sum\limits_{j \neq i} \text{cov}(X_i, X_j)}{\sum\limits_{j \neq i} \text{cov}(X_i, X_j)\text{max}}. \tag{4.3}$$

The H coefficient of a set of items can also be written as a normed covariance;

that is,

$$H = \frac{\sum_{i=1}^{I-1} \sum_{j=i+1}^{I} \text{cov}(X_i, X_j)}{\sum_{i=1}^{I-1} \sum_{j=i+1}^{I} \text{cov}(X_i, X_j)_{\max}}. \tag{4.4}$$

Alternatively, H can be written as a weighted sum of the items H_is, or the bivariate H_{ij}s (Mokken, 1971):

$$\begin{aligned} H &= \frac{\sum_{i=1}^{I-1} (\sum_{j=i+1}^{I} E_{ij}) H_i}{\sum_{i=1}^{I-1} \sum_{j=i+1}^{I} E_{ij}} \\ &= \frac{\sum_{i=1}^{I-1} \sum_{j=i+1}^{I} E_{ij} H_{ij}}{\sum_{i=1}^{I-1} \sum_{j=i+1}^{I} E_{ij}}. \end{aligned} \tag{4.5}$$

Mokken (1971, pp. 149-152) showed that the MHM implies that $0 \leq H_{ij} \leq 1$, $0 \leq H_i \leq 1$, and $0 \leq H \leq 1$. Thus, positive values of these coefficients are necessary for the MHM to hold. We restrict our attention to dichotomously scored items. The generalization of our methods to polytomous items (using Equations 4.2, 4.3, and 4.4) is straightforward.

Theoretically, a Mokken (1971) scale is defined as:

Condition 1 $\text{cov}(X_i, X_j) > 0$, for all $i \neq j$, and

Condition 2 $H_i \geq c$, for all i, where c is a user-defined constant between 0 and 1 (default, $c = 0.3$).

The first scaling condition, which can be restated as $H_{ij} > 0$, is necessary but not sufficient for the MHM to hold (Mokken, pp. 149-150[1]; also see Holland & Rosenbaum, 1986). The second scaling condition serves a practical purpose and allows the user to manipulate the minimum discrimination of items joined into scales. Given the choice of c, not all items may be scalable. The scalable items agreeing with the MHM do, however, contribute to the correct ordering of subjects on the latent variable measured by each scale (Grayson, 1988; Hemker et al., 1997). For the interpretation of the strength of a scale, Mokken (1971; p. 185) derived the following rules of thumb: $0.30 \leq H < 0.40$ constitutes a weak

[1]Mokken (1971) originally used correlations.

scale; $0.40 \leq H < 0.50$ a medium scale; and $H \geq 0.50$ a strong scale. Mokken considered $c = 0.30$ a reasonable minimal requirement for item quality. The appropriate value of c depends the researcher's purpose of scaling. When highly scalable (or, high discrimination) items are required, c needs to be high. For more information on the effect of c on dimensionality results, see Hemker et al. (1995), Molenaar and Sijtsma (2000), and Van Abswoude et al. (2004).

Having a set of items with high H coefficients (Equation 4.2, or 4.3) does not necessarily mean that the set is sensitive to a single latent trait. For example, for items driven by moderately correlated traits, the H coefficient may be high despite the fact that more than one trait is measured (e.g., Van Abswoude et al., 2004). On the other hand, Hemker et al. (1995) showed for polytomous items that the dominant dimensions of a data matrix may be found when different values of c are used for analyzing the same data. In fact, because the scale conditions are necessary but not sufficient for satisfying the MHM, one should also check the monotonicity of a scale. Let R_{-i} denote the total score on a set of items minus the score on item i. The program MSP then provides a tool to check the monotonicity of each item via nondecreasing $P[X_i = 1|R_{-i}]$ in R_{-i}, known as manifest monotonicity (Junker, 1993). Methods such as DIMTEST could be used in addition to MSP to ascertain that Mokken scales satisfy weak LI.

MSP uses a sequential clustering algorithm to select items into scales. Sequential item selection as defined by Mokken (1971) and as incorporated in the MSP method has the following stepwise procedure. Item selection starts by joining of the item pair (i, j) with the highest H_{ij}, under the restriction that it is significantly positive. This is the start set of the procedure. Then, out of all remaining items, that item i that yields the highest H with the already selected items is added to the start set under the following three restrictions: First, item i should have a positive covariance with each of the already selected items (see Condition 1); second, H_i with respect to the already selected items should be significantly positive; and third, the H_i, with respect to the already selected items, should satisfy $H_i \geq c$ (see Condition 2). This step is repeated until no item remains for which the scaling criteria are satisfied when they are added to the already selected set. Once this occurs, the first Mokken scale has been formed. If, after forming a scale, more than one item remains, the procedure is repeated to form a second, a third (and so on) Mokken scale. Details about significance testing, the treatment of ties, or other aspects of the sequential algorithm can be found in Mokken (1971) and Molenaar and Sijtsma (2000). More formally, the sequential clustering algorithm proceeds as presented below.

Typical solutions found with this sequential procedure will be illustrated by

Sequential algorithm:		
Repeat	Initial configuration	
	Repeat	Evaluate an unselected item w.r.t. a criterion
		Move best item if conditions are satisfied
	Until	Conditions not satisfied
Until	Condition not satisfied, or less than two items left	
Halt		

means of a small example using simulated data. Say, we have data on a linguistics test having items on the three topics 'grammar' (θ_1; 20 items), 'meaning' (θ_2; 10 items) and 'punctuation' (θ_3; 10 items). For such a test, one can easily imagine that the underlying abilities are correlated: we used r(θ_1,θ_2)=0.4, r(θ_1,θ_3)=0.2, and r(θ_2,θ_3)=0.2. In addition items may be sensitive to more than one trait. Then, starting out with the best item pair, two grammar items, using the sequential MSA method the following partitioning is found[2]: 25 items in scale 1 (θ_1, θ_2; $H = .51$); 5 items in scale 2 (θ_2; $H = .46$); and 10 items in scale 3 (θ_3; $H = .60$). If we deviate from the default setting of MSP and use the next best pair as the starting set (i.e., two meaning items) we find: 10 items in scale 1 (θ_2; $H = .53$); 20 items in scale 2 (θ_1; $H = .60$) and 10 items in scale 3 (θ_3; $H = .63$). The combination of dependence on the start set, and the inability to move items into better fitting clusters is the drawback of the sequential method in a nutshell [see Molenaar and Sijtsma (2000) for instructions how to cope with these issues in the current program MSP].

If the intention is to create a single reliable test (single scaling), then the user should choose an appropriate c, and the algorithm should *minimize the loss in H when as many items that satisfy the scale conditions are included in the scale as possible.* This means that out of the two solutions presented above, the default setting of MSP provides the preferred solution: a 25-item test sensitive to a mixture of traits. One should note, however, that the sequential algorithm may not have yielded the scale with the highest possible H given the obtained number of items, and the sequential algorithm provides no means to find out whether this is the case. Thus, hypothetically there could be a different set of 25 items with a higher H. For the data example presented here such a solution is unlikely. If a unidimensional test is preferred, stricter scale conditions should be used (i.e., a higher value for c).[3]

[2]Number of items, underlying dimensions, and scale H are presented for each scale.

[3]Using the sequential algorithm of MSP the preferred solution may also be found by increasing

When the goal is to create scales for each subability (multiple scaling), the first solution for the simulated data seems to be less appropriate. One may note that in this partitioning scale two is relatively small, whereas scale one is large and dimensionally heterogeneous. In addition, for every scale H is highest when items sensitive to θ_2 are joined in the same scale. In multiple scaling, the algorithm should *join each item into the scale it fits best and with which it satisfies the scale conditions.*[4] The solution that was obtained using a nonstandard start set is the best solution according to this definition.

The example illustrated that sequential MSP may not yield the preferred solution if our interest is in multiple scaling. It was also explained that in single scaling the preferred solution may not be obtained either. How typical is the presented example for empirical test construction situations? These problems can be expected to occur in scaling contexts where the underlying traits are significantly correlated (i.e., approximately $> .4$; e.g., Van Abswoude et al., 2004) or where items load on more than one trait. These conditions are realistic in many test data situations. In the next section, alternatives to the sequential algorithm are proposed that resolve these problems.

4.3 Alternative Clustering Methods

In this section, three new methods for Mokken scale analysis that might improve MSA's optimization in a single as well as in a multiple scale context are introduced. Because of this new approach, we need to define functions based on H that can be used to evaluate the quality of a partitioning consisting of items that are joined into one or more clusters. Each new function is called an objective function and its purpose is to find a scaling solution that maximizes its value such that, for example, the highest value of the H coefficient for all clusters is obtained simultaneously. The algorithms used to achieve this purpose allow single items to be moved to a different cluster where the fit may be better and allow multiple clusters to be formed simultaneously. In the next sections, three new objective functions that are based on H_{ij}, on H_i, and on H are introduced, and the way these objective functions are maximized is explained.

---

c and thereby making the scale conditions more restrictive (see Hemker et al., 1995). In our method it is not necessary to change the value of c. As a result, our scales may have more items (which may add to the reliability of the final test) than those obtained using the approach of Hemker et al. (1995).

[4] One can easily verify that if it is desirable to obtain disjoint multiple scales, the definition for a single scale presented above cannot be used.

4.3.1 Objective Functions Using the H Coefficient

Objective function O_1 was inspired by the work of Kim (1994) and Zhang & Stout (1999b). Their methods try to minimize a function of the conditional covariances between items. Objective function O_1 is similar to their function, but adapted for pairwise H_{ij} values (see Equation 4.2). It also similar to a proximity measure that was successful in finding the dimensionality using hierarchical clustering methods (Van Abswoude, Vermunt, Hemker, & Van der Ark, in press; chap. 2). The relationship between H_{ij}, H_i and H is the following: $\min(H_{ij}) \leq \min(H_i) \leq$ H $\leq \max(H_i) \leq \max(H_{ij})$ (e.g., Hemker et al., 1995; Mokken, 1971). Let $\eta_{ij} = 1$ if items i and j are in the same cluster k in partitioning $\mathcal{P}$, and $\eta_{ij} = -1$ otherwise. Then, O_1 is defined as

$$O_1 = \frac{2}{I(I-1)} \sum_{1 \leq i < j \leq I} \eta_{ij}(H_{ij} - c^*). \tag{4.6}$$

The idea behind this objective function is that its highest value will be obtained for the partitioning where items having the highest H_{ij}s are joined in the same cluster and those the lowest H_{ij} in different clusters. The partitioning that maximizes the objective function is referred to as $\mathcal{P}^*$. Multiplication with η_{ij} is incorporated in Equation 4.6 to encourage item pairs with a high H_{ij} to be joined into the same cluster and item pairs with a low or a negative H_{ij} to be split into different clusters. This is because the contribution of pair (i, j) to O_1 is positive if $H_{ij} - c^* > 0$ and $\eta_{ij} = 1$, and if $H_{ij} - c^* \leq 0$ and $\eta_{ij} = -1$. The contribution to O_1 is negative if $H_{ij} - c^* > 0$ and $\eta_{ij} = -1$, and if $H_{ij} - c^* \leq 0$ and $\eta_{ij} = 1$. Variations of O_1 can be obtained, not only by different choices of c^*, but also by changing the definition of η_{ij}. For example, for $\eta_{ij} = 0$ and items i and j not in the same cluster k, and $\eta_{ij} = 1$ otherwise, the objective function would only target the within-cluster scalability.

What is a reasonable c^*? Rewriting Equation 4.6 makes the effect of c^* on the final clustering solutions clearer:

$$O_1 = \frac{2}{I(I-1)} \sum_{1 \leq i < j \leq I} \eta_{ij} H_{ij} - \frac{2}{I(I-1)} \sum_{1 \leq i < j \leq I} \eta_{ij} c^*. \tag{4.7}$$

If $c^* = 0$, the sum most right of Equation 4.7 equals zero. Thus, $\mathcal{P}^*$ is found when all items, except those having many negative H_{ij}s with other items, are joined in one large set. With $c^* = 1$, which expresses the maximum value of H_{ij}, the sum most right of Equation 4.7 is maximized when the items are distributed in equal numbers over the K available clusters. The objective function is maximized when K equally large sets consist of item pairs that jointly yield the highest $\sum H_{ij}$

Table 4.1: *Effect of Using Different c^* Values in O_1 on Obtained Dimensionality Results*

		c^*			
Test Composition	discr.	0	0.3	1	$\bar{H}_{ij}$
15/15	hi	[30]	[30]	true	true
	mo	[30]	true	true	true
15/30	mo	[45]	[18/27]	[21/24]	[16/29]

Note. 'true' indicated that the simulated structure was obtained; number of items obtained in each cluster presented in brackets, otherwise.

(see Equation 4.7). This means that $c^* = 1$ only is appropriate when item sets are equal in size. Since researchers do not know the latent trait composition of their test, $c^* = 1$ is useless in practise. Choosing a fixed value for c^*, for example $c^* = 0.3$, is not advisable either, because the suitability of a c^* value may depend on the properties of the investigated items. A more reasonable value of c^* may be $c^* = 2/I(I-1)\sum_{i \neq j} H_{ij}$; that is, the average H_{ij} of all item pairs (denoted $\bar{H}_{ij}$). Kim (1994) used a similar adjustment in her objective function, although her intention was to correct for biased conditional covariance estimates. One can easily verify that if $c^* = \bar{H}_{ij}$ and all items are entered in one cluster, that $O_1 = 0$. This provides a benchmark against which other solutions can be compared.

By means of a few generated data examples, the appropriateness of the four c^* values (0, 0.3, 1, and $\bar{H}_{ij}$) for data reflecting two moderately correlated traits (i.e., 0.4) is investigated. The specific algorithms and model used for generating data are the same as in the main simulation study. The items sensitive to each underlying variable are highly (hi) or moderately (mo) discriminating and make up a short (15 items) or a long subtest (30 items). The results are presented in Table 4.1. It can be seen that if $c^* = 0$ all items were joined into one cluster, regardless of the type of data. Using $c^* = 0.3$ yields the simulated structure for moderate discrimination items but not for high discrimination items. This is because $H_{ij} > 0.3$ for most item pairs in the high item discrimination condition, and as a result items sensitive to different traits are joined into one cluster. As indicated in the table, using $c^* = 1$ only works for data having equal numbers of items for each trait (i.e., the total pool is split correctly), but not for unequal numbers. Using $c^* = \bar{H}_{ij}$ yields the simulated structure for equal numbers of items and has one misclassified item for unequal numbers. The use of $c^* = \bar{H}_{ij}$ in O_1 is investigated more extensively in the main study.

The second objective function, denoted as O_2, can be interpreted as the average H_i within clusters of a partition. The objective function is used to maximize the item scalability. The relationship between H_i and H is the following: $\min(H_i) \leq H \leq \max(H_i)$ (e.g., Hemker et al., 1995; Mokken, 1971). Before defining O_2, some additional notation is needed. Let k denote an arbitrary cluster of items ($k = 1, \cdots, K$) and let H_i^k be the H_i-value of item i with respect to the other items in cluster k. Let $\eta_i^k = 1$ if $i \in k$ at $\mathcal{P}$ (i.e., when item i is in cluster k), and $\eta_i^k = 0$, otherwise. The second objective function for evaluating a K-cluster partitioning $\mathcal{P}$ is

$$O_2 = I^{-1} \sum_{i=1}^{I} \sum_{k=1}^{K} \eta_i^k H_i^k. \tag{4.8}$$

We use normalizing constant I^{-1} and indicator η_i^k to make O_2 easily interpretable as the average H_i within clusters. Note that with O_2 all elements of a partitioning are evaluated and not just one element at a time as in the sequential MSP method. This means that O_2 can be used to search for that partitioning of items that produces the highest H_i for all items. This property may resolve the problems discussed for MSP. As one can observe, a constant c (see Mokken scale Condition 2) or any related constant like c^* in O_1 is not specified. One may further note that maximizing O_2 may not yield the same solution as maximizing H. We use O_2 because of its direct relationship to the second Mokken scaling condition.

Let H^k denote the total H of cluster k. The third objective function, denoted as O_3, equals the average within-cluster H:

$$O_3 = K^{-1} \sum_{k=1}^{K} H^k. \tag{4.9}$$

Out of the three presented objective functions, O_3 is most similar to MSP's original objective function.

The three objective functions presented above are similar, but clearly not the same. Objective functions O_1, O_2 and O_3 have in common that they use an average of the H_{ij}s. A difference is that O_1 uses the arithmetic mean of the H_{ij}s and O_2 and O_3 a weighted normalized sum of the H_{ij}s. Furthermore, O_1 targets both the within-cluster similarities and the between-cluster differences whereas O_2 and O_3 target only the within-cluster similarities.

4.3.2 NHCA Algorithm

A well known non-hierarchical clustering analysis (NHCA) algorithm is used to optimize O_1, O_2 and O_3. It is similar to the K-means algorithm (e.g., Berthold

& Hand, 1999). Let t ($t = 1, \cdots, T$), represent the iteration number. The NHCA algorithm is presented below.

NHCA Algorithm:	
t=0	
Initial configuration	
Repeat	Evaluate objective function
	Move an item to a cluster according to some criterion
	t=t+1
Until	Convergence
Halt	

In a NHCA, first an initial configuration is constructed. This means that each item i is assigned to its initial cluster k. At iteration t, the quality of the partitioning is evaluated using O_1, O_2 or O_3, and one item i is moved from a cluster k to another cluster k'. These steps are repeated until the process has converged.

The NHCA can be implemented in different ways. In our implementation, one item is moved at the time, but it would also have been possible to move more than one item per iteration. However, we choose not to move multiple items per iteration in our application because the values of the three objective functions are dependent on the number of items in each cluster because the average covariance decreases when more items are added to a cluster. We found that moving multiple items in one iteration step yielded instable results.

When applying the algorithm, we aim at finding the partitioning that yields the highest (or, the near highest) value of the objective function given all possible partitionings. One may note that the objective function is a discrete function of the partitioning and the objective function's value. As explained earlier, the solution space is only bound by the K investigated clusters (i.e., no Mokken scale conditions are imposed). The best solution is the one that maximizes the objective function. This solution is known as the global optimum solution. Frequently, however, this objective function is multi-modal; meaning that there are many local maxima and perhaps more than one interesting global maxima. In this chapter, the highest maximum that is obtained by running each of the algorithms is denoted the global maximum. It was also denoted the nearly globally optimal solution. Strictly speaking this solution is only global by approximation because not all possible solutions were investigated.

In general, because simple deterministic algorithms have a higher probability

to yield a local maximum, probabilistic algorithms are used in addition to deterministic ones. Examples of deterministic algorithms are the sequential clustering algorithm used for Mokken scale analysis (Molenaar & Sijtsma, 2000) and the hierarchical clustering algorithm (for applications, see Van Abswoude et al., in press; Roussos et al., 1998). Examples of probabilistic algorithms that may be used for MSA are: simulated annealing (e.g., Berthold & Hand, 1999), branch & bound algorithms (see Veldkamp, 2001, for an application to parametric IRT), genetic algorithms (e.g., Michalewicz, 1996), neural networks (see Swingler, 1996) and tabu search (e.g., Glover & Laguna, 1997).

For the new methods, a NHCA algorithm is used because it closely resembles MSP's original algorithm and because it has the potential to resolve the optimization problems discussed for the sequential approach. Another attractive property is that deterministic and stochastic elements can easily be incorporated. These elements influence the likelihood that a global solution is obtained and the speed of the algorithm. Deterministic and stochastic elements can be introduced into the NHCA at two occasions: at the initial configuration and at the move of a single item into a different cluster.

4.3.3 Initial Configuration

In the *random* initial configuration condition, items are randomly assigned to one of K clusters with equal probability (i.e., the default random number generator of Borland Pascal is used, version 7.0). Random initial configuration requires no additional information and thus is simpler than its deterministic counterpart. Methods that use a random start configuration may be repeated several times so that some may yield global optimal solutions (e.g, Michalewicz, 1996).

In the *non-random* initial configuration condition, the K-cluster partitioning obtained from a priori knowledge or obtained with another clustering method is used as an initial partition. One may use a sequential clustering procedure or a hierarchical clustering procedure, such as complete linkage, on the H_{ij}-matrix (Van Abswoude et al., in press), for this purpose. We used the default item selection procedure of MSP. The solution at the non-random start configuration may be close to the underlying dimensionality, and it can therefore be expected that few iterations will be needed to arrive at a final solution.

A practical complication is that sequential MSP does not necessarily yield the same number of clusters and may not use the same number of items as NHCA. There are different ways to remedy this. One approach is to try to obtain a K-cluster solution by manipulating constant c in sequential clustering. It may,

however, not be easy to find the desired c (it requires that the sequential method is run several times). In the simulation study, items from 'redundant' clusters and non-selected items were distributed over the K available clusters of NHCA. When sequential clustering yielded too few clusters, items from the largest sequential clusters were distributed over the K desired clusters. An effect of this reassigning of redundant clusters may be that somewhat more iterations in the move step are required than when c would have been manipulated to find K clusters.

4.3.4 Move an Item to a Cluster

In the non-random, or deterministic, condition, the item i is moved to that cluster k that yields the greatest improvement of the objective function. Methods that have such a deterministic move (i.e., hill-climbing methods) may only provide a local optimum value; therefore, the success of the method depends on the starting point of the algorithm.

In the random, or stochastic, condition there is a probability P_{it}^k that an item i will be moved to cluster k at iteration t. We use an adapted Metropolis procedure which is frequently used in simulated annealing (e.g., Berthold & Hand, 1999). This probability is based on the change in the value of the objective function when item i is moved to a cluster k at iteration t, denoted as ΔO_{it}^k. Note that if item i stays in the same cluster k, $\Delta O_{it}^k = 0$. Objective function O_{it}^k denotes one of the objective function O_1, O_2 and O_3. The probability that item i is moved to cluster k equals

$$P_{ikt} = \frac{[\exp(\Delta O_{it}^k)]^{t/I}}{\sum_{i=1}^{I}\sum_{k=1}^{K}[\exp(\Delta O_{it}^k)]^{t/I}}. \tag{4.10}$$

The denominator is used to normalize the probability distribution. Exponent (t/I) is added to make improvements (with respect to the value of the objective function) more likely for higher iterations. Which item i is moved to what cluster k subsequently depends on a randomly drawn number and the probabilities described above. If such a random-move procedure works well, the method should find the global solution each time it is run, making the repeated runs redundant.

4.3.5 Convergence

When moving items into clusters deterministically, the method stops when the objective function can no longer be improved. With a random component, it is less obvious when to stop the clustering process. We need a convergence rule to stop the iteration process. Figure 4.1 depicts the value of the objective function

as the iteration number increases for some binary multidimensional data. One can observe that for low iteration numbers, the objective function may increase or decrease. For $t \longrightarrow \infty$, the algorithm becomes similar to a deterministic algorithm where only improvements occur. In Figure 4.1, one can observe that the process does not become completely deterministic because the value of the objective function fluctuates between the best (global) solution and the next best solution. This is because in our implementation we move one item at every iteration step.

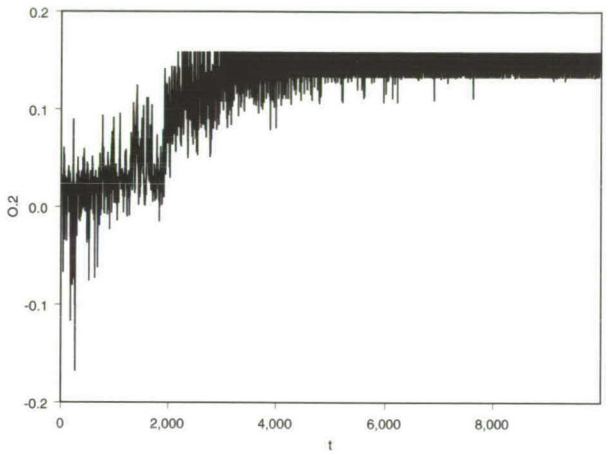


Figure 4.1: *Example of Convergence Process When Move is Random*

Different convergence rules can be used. We let the iteration process continue until a more or less stable result was obtained. In Figure 4.1 this is approximately after 6,000 iterations. We tried out different rules. By means of trial and error it was found that the following convergence rule produced stable results: obtain the same maximum value 100 times. This rule was used in the simulation study.

The Mokken scale conditions were not used in this study. In the appendix and in chapter 5 it is explained how the Mokken scale conditions may be incorporated into the new optimization method.

4.4 Simulation Study

The first goal of the simulation study was to investigate how successful the objective functions O_1, O_2 and O_3 are in assessing the underlying dimensionality

structure for different data matrices. For this purpose, the correspondence of the - by approximation - global solution and the underlying dimensionality of the data was compared. To obtain the global solution, all algorithms (i.e., different initial configurations and different moves) were run, and the highest value that was obtained was denoted "the global optimal value". This is the first part of the simulation study. The second research question was which algorithm can find the global maxima for O_1, O_2 and O_3. In answering this question the underlying dimensionality was ignored and only which algorithms yielded global solutions was assessed. This will be discussed in the second part of this section.

4.4.1 Model Used for Generating Data

For generating binary item scores, we introduce a model that can produce item response functions (IRFs) with highly varying shapes, thus approaching the flexibility of nonparametric IRT as much as possible. The flexibility of the model lies in the fact that an item can have more than a single inflection point (also, see Douglas & Cohen, 2001; Samejima, 2000). The model used for generating binary responses is a multidimensional IRT model that consists of a mixture of the items step response functions (of polytomous items) that satisfy the multidimensional two-parameter logistic model (M2-PLM; Birnbaum, 1968; Reckase, 1997). The item step response functions that define the mixture model are referred to as 'components'.

Before we define the model, we need to introduce some notation. Let q ($q = 1, \cdots, Q$) represent the mixture components of item i, let α_{iqd} denote the discrimination parameter of component q on trait d ($d = 1, \cdots D$) for item i, and let δ_{iqd} denote the component-specific difficulty parameter for item i. The component-specific difficulty may be interpreted as the location where the component discriminates most.

The mixture model is defined by

$$P(X_i = 1|\theta) = \sum_{q=1}^{Q} \frac{\exp[\sum_{d=1}^{D} \alpha_{iqd}(\theta_{pd} - \delta_{iqd})]}{1 + \exp[\sum_{d=1}^{D} \alpha_{iqd}(\theta_{pd} - \delta_{iqd})]}. \tag{4.11}$$

The advantage of this mixture model is that nondecreasing IRFs with many different shapes can be obtained. Increasing the number of components in the model generally means that more inflection points are added to the IRF. Increasing the α_{iqd}s means that the local increases, or bumps, in the IRF become steeper. Increasing the α_{iqd}s does not, however, unequivocally manipulate the overall discrimination of an item. However, because an IRF differs locally in steepness,

increasing α_{iqd} may also increase the overall discrimination of an IRF. The item discrimination can more directly be manipulated via the δ_{iqd}s; that is, increasing the variance of the δ_{iqd}s within an item lowers the discrimination of an item.

In the simulation study, depending on the particular cell in the design, the parameters of the mixture model had different values. The values of item component parameters were first generated from a distribution and subsequently fixed so that the results of the various conditions of the design became comparable. This means that the values of α_{iqd} and δ_{iqd} are different between items and between components, but the item properties were the same between equivalent conditions of the design.

Each of the IRFs has five components. Depending on the particular factor of the design, the values of the component-specific discrimination parameters were drawn from the following distributions: *high* from $U(4;8)$; *medium* from $U(1.75;4)$; *low* from $U(0.25;1.25)$; or they were zero when an item was not sensitive to a particular trait beyond the sensitivity to this trait caused by the correlations between the latent traits. The labels high, medium and low are used to refer to different levels of item discrimination. The component-specific difficulty parameters were either drawn from a relatively broad range or from a narrow range to manipulate the overall discrimination of an item. To obtain variation in item difficulties within each level of overall discrimination, we drew δ_{iqd}s from different uniform distributions. For a *moderate* item discrimination set having 15 items, component-sensitive difficulties of five items were drawn from $U(-3;3)$, five from $U(-4;2)$ and five from U(-2;4). For *high* item discrimination, component-sensitive difficulties were drawn from $U(-1.5;1.5)$, $U(-2.5;0.5)$ and $U(-0.5;2.5)$.

The effects of the component-sensitive parameters (i.e., three levels of component-sensitive discrimination and two levels of component-sensitive difficulty) on the shape of the IRF will be illustrated in two ways. First, we investigated the effect in comparison with 2-PLM parameters. We did this by searching which 2-PLM approximates the data generated with the five-component mixture model best. We used LEM (Vermunt, 1997) for this purpose. Second, we present plots of IRFs for the different conditions. For simplicity, in both illustrations unidimensional items were used; all other properties of the items were the same as in the simulation study. Because the values of the parameters were fixed across conditions, this meant that the δ_{iqd}s of items were exactly the same across different levels of component-sensitive discriminations, making it possible to assess the effect of α_{iqd} without the influence of δ_{iqd}. The reverse (i.e., α_{iqd} fixed and δ_{iqd} free) was true for δ_{iqd}.

The parameter estimates obtained with LEM are presented in Table 4.2 for

Table 4.2: *Minimum and Maximum Value of the 2-PLM Parameters Item Discrimination and Item Difficulty for Data Generated Using Two Ranges of Component Difficulty (δ_{iqd}) and Three Levels of Component Discrimination (α_{iqd})*

	Level of α_{iqd}		
Range of δ_{iqd}	High	Medium	Low
	Item Discrimination		
Small	1.36–4.10	1.27–2.28	0.49–0.89
Large	0.70–1.54	0.64–1.58	0.44–0.87
	Item Difficulty		
Small	-1.26–1.32	-1.32–1.22	-1.24–1.35
Large	-1.32–1.68	-1.23–1.36	-.98–1.41

the three levels of component-discrimination (high, medium and low) and the two ranges of component-difficulty (large and small range). It can be observed from the upper half of Table 4.2 that the item discrimination was substantially higher when the range of component-sensitive difficulties was small (first row) rather than large (second row). Thus, the manipulation of the item discrimination via the component-sensitive difficulties was successful. Decreasing component-sensitive discrimination (columns 1-3), in addition to decreasing 'bumps' in the IRFs, reduced item discrimination. This was especially found in item sets having a small range of component-sensitive difficulties. The values of the difficulty parameters presented in the lower half of Table 4.2 indicate that drawing δ_{iqd} from multiple distributions as was explained earlier produced sets with items that varied in popularity.

In Figure 4.2 three typical IRFs for each of the three levels of component discrimination and the two levels of component difficulty are presented. The horizontal axis depicts the θ value and the vertical axis $P(X = 1|\theta)$. The plots of the IRFs illustrate the effects of parameter values presented in Table 4.2 on the shape of the curves.

4.4.2 Design

Retrieving the Dominant Underlying Dimensionality

To answer the research question regarding the successfulness of the objective functions, we used data matrices having 15 items per latent trait, and 2,000 responses per item. Throughout the study, we used 2-dimensional standard normally dis-

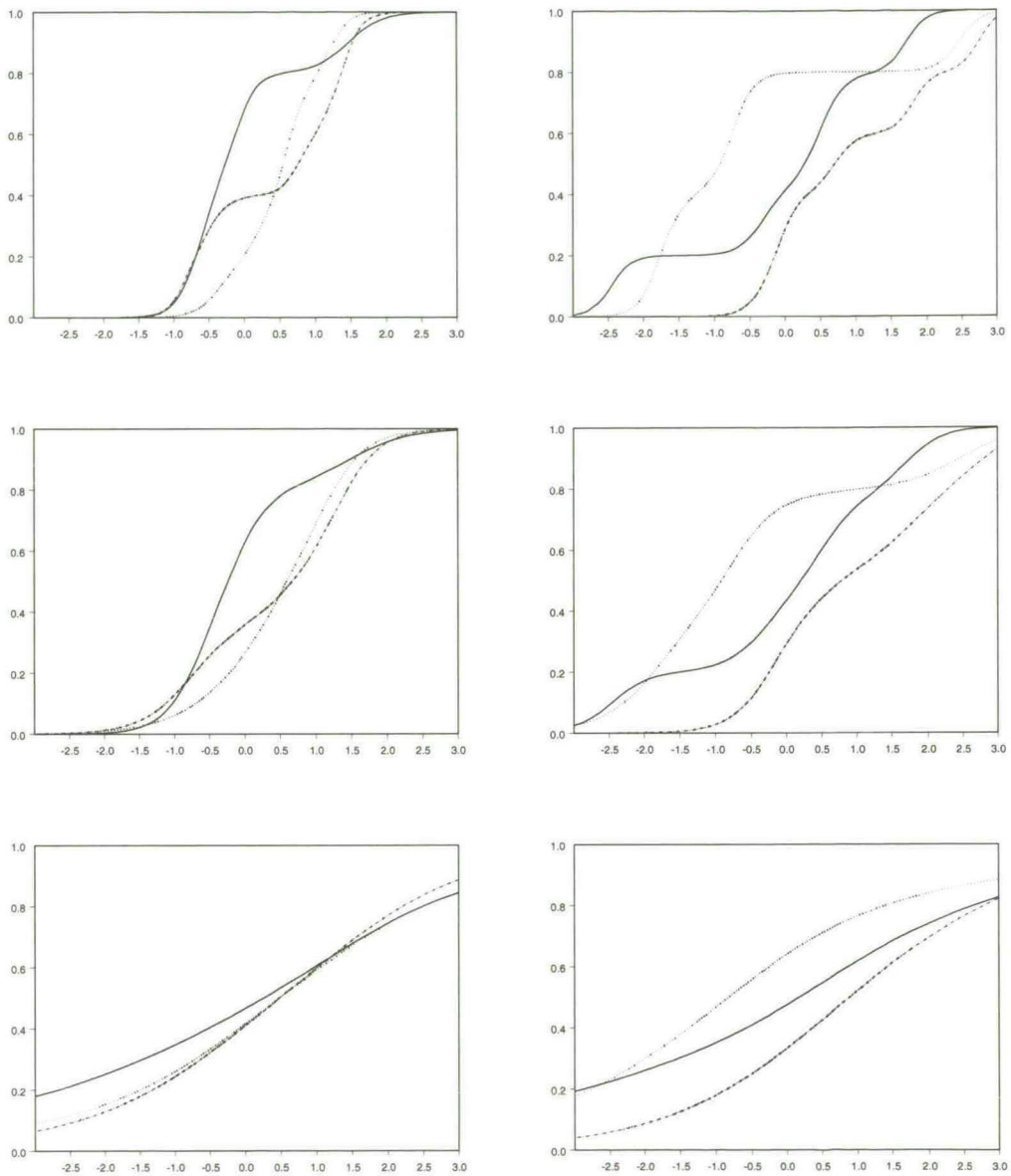


Figure 4.2: *Examples of Three Items Having High, Medium or Low Component Discrimination α_{iqd} (Depicted Vertically) With Small Range (First Column) and Large Range of Component-Difficulty δ_{iqd} (Second Column) (Rows and Columns Were Reversed Compared to Table 4.2 for Display Purpose)*

tributed latent traits (θ_1 and θ_2), and 5 components (in Equation 4.11). In order to ensure stability of the results, we replicated each cell 10 times. The design comprised the completely crossed factors 'Correlation Between Traits' (three levels), 'Structure' (four levels), and 'Item Discrimination' (2 levels), which yielded a $4 \times 3 \times 2$ design. For a part of the design we also varied the 'Numbers of Items per Trait' (two levels), and the 'Calibration Size' (three levels).

The three levels of Correlation Between Traits (ρ) were 0.1, 0.4, 0.7. An increase of ρ, decreases the differences between the responses to items that are sensitive to different traits and thus the more difficult it becomes to partition items into subsets that correspond to the simulated dimensionality. The extremes $\rho = 0.0$ (i.e., no correlation) and $\rho = 1.0$ (i.e., a unidimensional model fits the data best) were excluded because they provide little challenge for the methods. We chose to use the values 0.1, 0.4 and 0.7 because in earlier studies (Van Abswoude et al., 2004) for high discrimination data MSP could find the dimensionality of the data for correlations up to $\rho = 0.4$, and we would like to know whether better results can be obtained using the new methods provided in this chapter.

The four levels of Structure were conditions AS1, AS2, AS3 and AS4 (AS stands for approximate simple structure; as used by Stout (2002)). In condition AS1, items were constructed to be highly discriminating on one latent trait (i.e., the intended trait). Discrimination with respect to the other latent trait (i.e., the unintended latent trait) was entirely due to the correlation between the traits. In AS2, items discriminated highly with respect to their intended trait, and lowly with respect to their unintended trait. Figure 4.3 depicts an item response surface used in the AS2 condition. In condition AS3 items discriminated highly with respect to their intended trait and medium with respect to their unintended trait. In condition AS4 items had medium discrimination with respect to their intended trait and low discrimination with respect to their unintended trait. Thus, in AS4 the overall discrimination was different compared to AS3, but the deviations from simple structure in both levels were comparable. We expected that the simulated dimensionality structure in AS1 was easiest to recover, followed by AS2, and then by AS3 and AS4 alike. The four levels of Structure were manipulated via the component-specific discriminations and their specific values were presented earlier.

The two levels of Item Discrimination were 'high' and 'moderate'. The overall discrimination of an IRF was manipulated via the dispersion of the δ_{iqd}s. Although there is a relationship between the α_{iqd}s (component-specific discriminations) and the α_{id}s (item discriminations), the α_{iqd}s were held constant between the two levels of the factor Item Discrimination. It may be noted that the factor Item

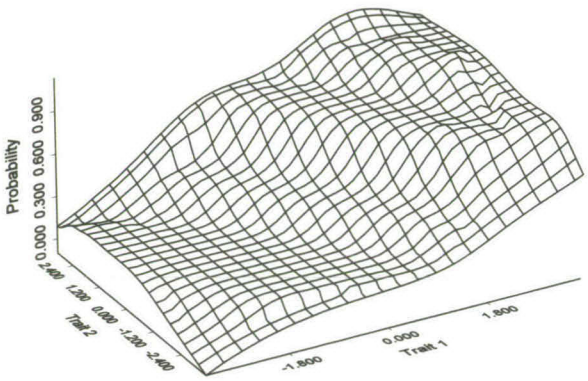


Figure 4.3: *Item Response Surface of an Item that Discriminates Highly With Respect to θ_1 (T1) and Lowly With Respect to θ_2 (T2)*

Discrimination was manipulated independently from the factor Structure. Thus, within one level of Structure (say, AS1) the item discrimination may be high or moderate. The parameter values were presented earlier. Thus, although the discrimination of items is varied for the factors Structure and Item Discrimination alike, the differences are clearer between conditions of the factor Item Discrimination.

The two levels of Numbers of Items per Trait were 'equal numbers' (i.e., 15 items sensitive to θ_1 and 15 items to θ_2) and 'unequal numbers' (i.e., 15 items sensitive to θ_1 and 30 items to θ_2). Numbers of Items per Trait was included as a design factor because it has been shown to have an effect on finding the dimensionality of a set of items using LI-based methods (Van Abswoude et al., 2004). The effect of this factor was investigated for a few cells.

The three levels of Sample Sizes we used were 'small' (i.e., 200 subjects), 'medium' (i.e., 2,000 subjects), and 'large' (i.e., 10,000 subjects). It was expected that the results using small sample sizes are less stable than the results for medium or large sample sizes. The effect of this factor was only investigated for a few cells.

Dependent Variables Judgement of the success of the methods was based on two criteria. The first criterion was whether the simulated structure was found, meaning that the items were split according to their underlying trait struc-

ture. The retrieved clustering solution has the following general structure $[K : I^1; I^2; \cdots; I^K]$, where K denotes the number of obtained clusters, and $I^1, \cdots, I^K$ denotes the number of items retrieved for clusters 1 through K. When two adjacent clusters (e.g., I^1 and I^2) are separated by a semicolon this indicates that no classification errors were made (i.e., all items are sensitive to the same intended trait); when separated by a comma, the two sets are sensitive to the same underlying trait; and when separated by a slash, some items are entered into a cluster sensitive to a different trait. When a method yields the simulated structure, this is referred to as 'the true dimensionality'. This is an observation, not an interpretation. For example, when two latent variables correlate .95 and partitioning [2:15;15] is found, from a substantive point of view one may prefer a solution like [1:30].

The second criterion was the value of the objective function at the global maximum solution. Note that the value of the objective function is an indication of the strength of the clusters that were found.

Performance of the Algorithms

The second research question relates to the elements of the clustering method that are responsible for finding the global optima. For this purpose, we compared the NHCA methods with respect to their ability to find the global maxima for different types of data. We used only a part of the total design. We investigated the effect of Structure (four levels), and Correlations Between Latent Traits (three levels) on the number of times the global optimum was found for the different NHCA algorithms. The four algorithms (i.e, two initial configurations and two move processes) were investigated for O_1, O_2 and O_3 each. The data were generated using the five-component mixture model with moderately discriminating items. The probabilistic methods were run 10 times.

Dependent Variables The success of the algorithms was judged by using as a criterion the number of replications that resulted in a global optimal solution. We also determined the average and standard deviation of the number of runs that produced the global optimum solution for 10 replicated data matrices. The average number of iterations needed to find the global optima was the second dependent variable.

4.5 Results

4.5.1 Retrieving the Dominant Underlying Dimensionality

High Item Discrimination

Table 4.3 shows the dimensionality results using sequential clustering and NHCA for data generated with the five-component mixture model and high discrimination items (narrow distributions of the component-difficulties). For the sequential MSA method (in short: Sequent), we only present the clustering solution and not the values of the objective functions O_1, O_2 and O_3 because their values may be misleading when the number of items and/or the number of clusters is not the same as for NHCA. For non-hierarchical clustering, we present the global maxima of the objective functions, as well as the value of the objective function for the true dimensionality (presented within parenthesis). In Table 4.3, the label 'true' is used to denote that the simulated dimensionality was retrieved (i.e., the global solution is the same as the true dimensionality), otherwise the retrieved global clustering solution is printed.

As Correlations Between Latent Traits (ρ) increased the following effects can be observed in Table 4.3: Sequent tended to collect all items in one large cluster; O_1 and O_2 found the underlying dimensionality; and O_3 tended to split the total set in one item pair versus the rest. When loadings on unintended traits increased (conditions AS3 and AS4) similar effects were observed. For low discrimination, Sequent and to a lesser extent O_1 did not yield the true dimensionality, whereas O_2 did most of the time. The value of the objective function increased with increasing ρ, increasing loading on unintended traits, and increasing Item Discrimination. In general, O_1 and O_2 performed better than Sequent, and O_3 performed worse.

The explanation of Sequent's results is that for high ρ and deviation from AS1 most items satisfied the Mokken scale conditions for the first scale, which means that they were all collected into the first scale (see Table 4.3). This effect was stronger in the high discrimination condition. With high ρ and moderate discrimination, some items could not satisfy the scale conditions of the large cluster, but satisfied the scale conditions when a new scale was formed (an example can be found in a later table) or were not scalable at all (e.g., AS4 and $\rho = 0.1$ in Table 4.3).

The difference between the the global maxima and the value of the objective function at the true dimensionality partitioning (i.e., the global maximum is equal or larger than the maximum at the true dimensionality) informs us about the suitability of the method for dimensionality analysis. When there is no difference, or

Table 4.3: *Results of the Restricted Sequential MSP and the Unrestricted New MSA Methods Using Objective Functions O_1, O_2 and O_3 on Retrieved Dimensionality for High Discrimination Data*

		Sequent ($c = 0.3$)	NHCA					
			$O_1(H_{ij} - \bar{H}_{ij})$		O_2 (H_i^k)		O_3 (H^k)	
Structure	ρ	Clust.	Value	Clust.	Value	Clust.	Value	Clust.
AS1	.1	true	.290	true	.573	true	.643 (.561)	[2/28]
	.4	[14/16]	.193	true	.560	true	.680 (.548)	[2/28]
	.7	[30]	.089	true	.563	true	.728 (.552)	[2/28]
AS2	.1	true	.210	true	.529	true	.658 (.514)	[2/28]
	.4	[29]	.127	true	.543	true	.699 (.528)	[2/28]
	.7	[30]	.067	true	.562	true	.738 (.551)	[2/28]
AS3	.1	[28]	.070 (.065)	[5/25]	.471	true	.692 (.451)	[2/28]
	.4	[30]	.068 (.039)	[6/24]	.500	true	.717 (.482)	[2/28]
	.7	[30]	.067 (.023)	[6/24]	.542 (.540)	[15/15]	.746 (.524)	[2/28]
AS4	.1	[14;14]	.110	true	.404	true	.627 (.390)	[2/28]
	.4	[28]	.071	true	.442	true	.675 (.428)	[2/28]
	.7	[30]	.046 (.034)	[6/24]	.478	true	.700 (.466)	[2/28]

Note. 'Clus.' represents the clustering solution of a method. 'True' denotes that the maximum value of the objective function was found at the simulated partitioning; the obtained partitioning is presented in notation between brackets, otherwise. Summary bracket notation: ';' separates dimensionally different sets of items; ',' separates dimensionally similar sets; and '/' separates dimensionally mixed sets. 'Value' denotes the objective functions' (near) global value and (if different) the value at the true dimensionality (within parenthesis). Global Maximum value and partitionings were obtained by running all algorithms.

the difference is small, the method appears to be suitable for dimensionality assessment, and the method appears to be unsuitable otherwise. For O_1 and O_2 these values were the same or almost the same (except for $\rho = 0.7$ and AS3). When the differences are small, this may mean that the data have more than one reasonable maximum, one of which but not the highest is found at the partitioning that is regarded as the true dimensionality. Obtaining partitioning solutions under these conditions may be highly susceptible to sample fluctuation. The values for O_3 were far apart. Objective function O_3 was not successful in finding the underlying dimensionality, because the compensation by means of a low H contribution to O_3 for the remaining items when one item pair yields a high H was not enough. We need to restrict the solution space (i.e., add Mokken scale conditions to the problem) more in order to find the true dimensionality.

The interpretation of the O_1 values is not so clear-cut due to constant c^*. Low values of O_1 occur when the pairwise H_{ij}s do not deviate much from their mean value. This may have different causes: items discriminate weakly, clusters correlate highly, or items load highly on each latent trait. To interpret the maximum values of O_2 (i.e., the average within-cluster H_i) we could use Mokken's rules of thumb. There are clear distinctions between H and O_2: O_2 can be seen as an average unweighed H over K clusters (see Equation 4.8). However, there are enough similarities that make the application of the rules defensible. When using Mokken's rules, we observe that the clusters obtained for high discriminating items were strong. Interpreting the maximum O_3 values using Mokken's rules of thumb indicates that the average cluster is strong.

As ρ increased and as the item loadings on unintended traits increased, it became more difficult to retrieve the simulated dimensionality structure and the methods became more susceptible to capitalization on chance.

High Item Discrimination: Replicated Data

To get some idea about the stability of the high item discrimination results, the analyses were repeated 10 times. Table 4.4 shows that the results were not sensitive to sampling fluctuation. The maximum values of the objective function did not change much between replications. In AS3, O_2 was maximized at the true partitioning in some replications, and at another partition in other replications (see Table 4.4).

Table 4.4: *Results of the Unrestricted New MSA Methods Using Objective Functions O_1, O_2 and O_3 Concerning Retrieved Dimensionality for Ten Replicated Data Matrices Having High Discrimination Items*

		NHCA					
		$O_1(H_{ij} - \bar{H}_{ij})$		O_2 (H_i^k)		O_3 (H^k)	
Struc.	ρ	av. (sd.)	# t.	av. (sd.)	# t.	av. (sd.)	# t.
AS1	.1	.288 (.008)	10	.556 (.009)	10	.403 (.018)	0
	.4	.192 (.007)	10	.557 (.006)	10	.392 (.037)	0
	.7	.096 (.006)	10	.559 (.009)	10	.427 (.021)	0
AS2	.1	.203 (.005)	10	.523 (.008)	10	.360 (.018)	0
	.4	.128 (.007)	10	.538 (.008)	10	.401 (.016)	0
	.7	.064 (.003)	10	.555 (.007)	10	.429 (.016)	0
AS3	.1	.072 (.004)	0	.461 (.013)	9	.340 (.016)	0
	.4	.070 (.003)	0	.500 (.011)	10	.388 (.025)	0
	.7	.065 (.003)	0	.543 (.008)	4	.431 (.013)	0
AS4	.1	.109 (.006)	10	.417 (.009)	10	.340 (.019)	0
	.4	.068 (.003)	10	.452 (.008)	10	.369 (.027)	0
	.7	.048 (.003)	0	.477 (.006)	10	.394 (.025)	0

Note. 'av.(sd.)' denote the average and standard deviation of the maximum objective function for ten replicated data matrices. '# t.' represents the number of correct partitions for the ten replicated data matrices.

Moderate Item Discrimination

Table 4.5 shows the dimensionality results of Sequent, O_1, O_2, and O_3 for moderately discrimination items.[5] It can be seen in Table 4.5 that the global objective function values were lower and the solution corresponding to the true dimensionality was found less often than for high item discrimination data. This result was found because the H coefficient is sensitive to the discrimination of items. The effects of the other design factors were similar to the effect of high discrimination items.

[5]One may recall that the average item discrimination between the two levels of the factor Item Discrimination is not the same and by implication, the average item discrimination for each Structure condition (AS1, AS2, AS3, or AS4) is differed between Tables 4.3 and 4.5.

Table 4.5: *Results of the Restricted Sequential MSP and the Unrestricted New MSA Methods Using Objective Functions O_1, O_2 and O_3 Concerning Retrieved Dimensionality for Moderate Discrimination Data*

		Sequent ($c = 0.3$)	NHCA					
			O_1 ($H_{ij} - \bar{H}_{ij}$)		O_2 (H_i^k)		O_3 (H^k)	
Structure	ρ	Clust.	Value	Clust.	Value	Clust.	Value	Clust.
AS1	.1	[2,9;10]	.129	true	.256	true	.383 (.256)	[2/28]
	.4	[9;9]	.078	true	.246	true	.417 (.245)	[2/28]
	.7	[9;2,9]	.050 (.041)	[7/23]	.260	true	.426 (.258)	[2/28]
AS2	.1	[2,6;2,5]	.080	true	.217	true	.377 (.216)	[2/28]
	.4	[2,7;9]	.054	true	.226	true	.383 (.226)	[2/28]
	.7	[8;2,8]	.044 (.027)	[8/22]	.241	true	.412 (.241)	[2/28]
AS3	.1	[2,4;2,5]	.050 (.018)	[7/23]	.179 (.177)	[12/18]	.364 (.178)	[2/28]
	.4	[2/2/2/9]	.051 (.014)	[7/23]	.216 (.211)	[13/17]	.388 (.211)	[2/28]
	.7	[2/2/2/12]	.053 (.007)	[6/24]	.243 (.229)	[13/17]	.443 (.229)	[2/28]
AS4	.1	[2,2,3;3,4]	.047	true	.193	true	.324 (.192)	[2/28]
	.4	[2,2,5;3,5]	.045 (.033)	[7/23]	.211	true	.334 (.210)	[2/28]
	.7	[2/2/2/5/8]	.042 (.017)	[7/23]	.225 (.224)	[12/18]	.418 (.223)	[2/28]

Note. 'Clus.' represents the clustering solution of a method. 'True' denotes that the maximum value of the objective function was found at the simulated partitioning; the obtained partitioning is presented in notation between brackets, otherwise. Summary bracket notation: ';' separates dimensionally different sets of items; ',' separates dimensionally similar sets; and '/' separates dimensionally mixed sets. 'Value' denotes the objective functions' (near) global value and (if different) the value at the true dimensionality (within parenthesis). Global Maximum value and partitionings were obtained by running all algorithms.

Moderate Discrimination: Replicated Data Table 4.6 presents the moderate discrimination results for the 10 replicated data matrices. It can be observed that there was little sample fluctuation. In Table 4.4 we saw sample fluctuation in condition AS3, but in Table 4.5 we mainly see sample fluctuation for AS2 (hi/lo) and AS4 (me/lo).

Table 4.6: *Results the Unrestricted New MSA Methods Using Objective Functions O_1, O_2 and O_3 Concerning Retrieved Dimensionality for Ten Replicated Data Matrices Having Moderate Discrimination*

		NHCA					
		$O_1(H_{ij} - \bar{H}_{ij})$		$O_2\ (H_i^k)$		$O_3\ (H^k)$	
Struc.	ρ	av. (sd.)	# t.	av. (sd.)	# t.	av. (sd.)	# t.
AS1	.1	.125 (.005)	10	.257 (.004)	10	.403 (.018)	0
	.4	.085 (.004)	10	.253 (.004)	10	.371 (.044)	0
	.7	.047 (.002)	1	.253 (.005)	10	.383 (.047)	0
AS2	.1	.078 (.004)	10	.220 (.006)	10	.315 (.036)	0
	.4	.054 (.003)	10	.230 (.005)	10	.364 (.042)	0
	.7	.050 (.003)	0	.247 (.005)	8	.386 (.042)	0
AS3	.1	.052 (.003)	0	.177 (.006)	0	.297 (.040)	0
	.4	.055 (.003)	0	.211 (.007)	0	.340 (.034)	0
	.7	.059 (.005)	0	.244 (.004)	0	.414 (.030)	0
AS4	.1	.047 (.003)	0	.187 (.005)	10	.309 (.039)	0
	.4	.042 (.002)	10	.212 (.005)	8	.345 (.037)	0
	.7	.046 (.002)	0	.231 (.006)	3	.381 (.026)	0

Note. 'av. (sd.)' denote the average and standard deviation of the maximum objective function for ten replicated data matrices. '# t.' represents the number of correct partitions for the ten replicated data matrices.

Moderate Discrimination: Numbers of Items and Sample Size Table 4.7 shows the results for varying the factors Number of Items and Sample Size for moderately discriminating items. These data were simulated under the AS2 condition. For Sequent, O_1, O_2, and O_3 the results of 10 replicated data matrices are presented.

Table 4.7 shows that the numbers of items did not influence whether Sequent found the true dimensionality. For unequal numbers of items per trait, Sequent generally produced more clusters than for equal numbers of items (not shown in Table 4.7). The results of O_1 were worse in the unequal numbers of items

Table 4.7: *Results of the Unrestricted New MSA Methods Using Objective Functions O_1, O_2 and O_3 Concerning Dimensionality Results for Different Numbers of Items per Trait and Number of Respondents for Moderate Discrimination Data*

		Sequent	NHCA					
		$(c = 0.3)$	$O_1(H_{ij} - \bar{H}_{ij})$		$O_2\ (H_i^k)$		$O_3\ (H)$	
	ρ	# True	av.(sd.)	# True	av.(sd.)	# True	av.(sd.)	# True
Default								
	.1	0	.078 (.004)	10	.220 (.006)	10	.315 (.036)	0
	.4	0	.054 (.003)	10	.230 (.005)	10	.364 (.042)	0
	.7	0	.050 (.003)	0	.247 (.005)	8	.386 (.042)	0
Numbers of items per trait								
$[2:15;30]$	.1	0	.077 (.005)	10	.219 (.009)	10	.379 (.020)	0
	.4	0	.052 (.010)	1	.228 (.004)	10	.409 (.018)	0
	.7	0	.030 (.013)	0	.243 (.006)	7	.440 (.022)	0
Sample size								
200	.1	0	.075 (.007)	2	.218 (.021)	4	.463 (.038)	0
	.4	0	.058 (.008)	1	.227 (.024)	3	.466 (.037)	0
	.7	0	.060 (.006)	0	.250 (.016)	0	.533 (.045)	0
$10,000$	.1	0	.079 (.002)	10	.221 (.003)	10	.358 (.010)	0
	.4	0	.053 (.002)	10	.232 (.003)	10	.384 (.013)	0
	.7	0	.050 (.001)	0	.241 (.004)	10	.414 (.009)	0

Note. 'av.(sd.)' denote the average and standard deviation of the maximum objective function for ten replicated data matrices. '# true' represents the number of correct partitions for the ten replicated data matrices.

condition than in the equal numbers of items condition, but they turned out to be better than expected. Evidently the effect of having clusters that are not equal in strength (i.e., due to unequal numbers of items or due to unequal item discrimination between clusters) is not so large for two-cluster data. The results of O_2 and O_3 with unequal numbers of items were not notably different from the equal numbers condition.

There is some sample fluctuation in Sequent's results, but not to the extent that simulated partitionings were retrieved in one condition and not in the other. Table 4.7 shows that the values of the objective functions varied more for small samples and the true dimensionality was found less often. This was not surprising because small differences in item scores can have large effects in small samples.

4.5.2 Performance of the Algorithms

The results of the new methods presented in Table 4.3 up to Table 4.7 were obtained by making use of all NHCA algorithms. Only the results that corresponded with maximum values of the objective functions were presented. In this section, the goal is to find out which algorithm was best in finding these global solutions. The algorithm that had the highest probability of finding the global solution is seen as the best algorithm. When several algorithms performed equally well, we prefer the one that finds the global optimum within the smallest number of iterations.

Table 4.8 presents the performance of the algorithms used for O_1 and O_2. We did not include O_3 because it was not very successful in unrestricted dimensionality assessment. The algorithms are abbreviated in the following way: SEQ and RAN1 denote the sequential and random initial configurations; and DET and RAN2 denote the deterministic and random moves. The four combinations of the two initial configuration possibilities and the two move possibilities yield the four algorithms used in this chapter: these are, RAN1&DET, RAN1&RAN2, SEQ&RAN2, and SEQ&DET. For each algorithm, we present: the number of times that, out of ten replications, the highest (global) maximum was reached (denoted # G.); the average (and standard deviation) number of runs that yielded global values over ten replications (denoted [denoted av.(sd.)]; not for SEQ & DET); and the average number of iterations needed to obtain the global solution for the first time (denoted $\bar{t}$).

Table 4.8: *Efficiency of Algorithms for O_1 and O_2*

		Initial: Random (RAN1)						Sequential (SEQ)				
		Move: DET			RAN2			DET		RAN2		
Structure	ρ	G	[av.(sd.)]	$\bar{t}$	G	[av.(sd.)]	$\bar{t}$	G	$\bar{t}$	G	[av.(sd.)]	$\bar{t}$
						O_1 ($c^* = H_{ij} - \bar{H}_{ij}$)						
AS1	.1	10	9.1 (1.0)	14	10	10 (0.0)	2,166	10	7	10	9.2 (0.9)	2,215
	.4	10	9.4 (1.0)	14	10	9.7 (0.7)	3,143	10	7	10	9.7 (0.7)	3,116
	.7	10	7.3 (2.1)	13	10	9.3 (1.3)	5,698	1	8	10	9.5 (0.8)	5,683
AS2	.1	10	9.2 (1.1)	14	10	9.5 (0.5)	3,453	10	9	10	9.3 (0.7)	3,385
	.4	10	5.0 (1.7)	13	10	10 (0.0)	5,026	10	9	10	9.9 (0.3)	5,143
	.7	10	8.8 (1.7)	14	10	8.5 (1.5)	5,357	4	11	10	8.3 (1.8)	5,361
AS3	.1	10	9.6 (1.0)	14	10	9.1 (0.7)	5,217	9	11	10	9.4 (0.7)	5,104
	.4	10	9.4 (1.1)	14	10	8.2 (1.5)	4,793	9	16	10	8.7 (0.7)	4,715
	.7	10	9.6 (1.3)	14	10	7.6 (1.7)	4,562	10	11	10	7.8 (1.9)	4,675
AS4	.1	10	5.9 (1.9)	13	10	10 (0.0)	5,654	4	11	10	9.8 (0.6)	5,682
	.4	10	8.7 (1.5)	14	10	9.1 (0.7)	6,380	6	11	10	8.8 (1.0)	6,446
	.7	10	8.8 (2.2)	14	10	8.8 (1.5)	5,858	8	15	10	8.8 (1.1)	5,667

Note. Number of Global Maxima out of Ten Replications (G);
Average (and Standard Deviations) of Reported Global Maxima ([av.(sd.)]) and
Average Number of Iterations ($\bar{t}$) Over Ten Runs of the Algorithms and
Ten Replicated Data Matrices Having Moderate Discrimination Items

Table 4.8: (continued)

		Initial: Random (RAN1)						Sequential (SEQ)				
		Move: DET			RAN2			DET		RAN2		
Structure	ρ	G	[av.(sd.)]	$\bar{t}$	G	[av.(sd.)]	$\bar{t}$	G	$\bar{t}$	G	[av.(sd.)]	$\bar{t}$
						O_2 (H_i^k)						
AS1	.1	10	[10 (0.0)]	14	10	[9.0 (1.5)]	2,264	10	7	10	[8.9 (1.0)]	2,219
	.4	10	[10 (0.0)]	14	10	[9.3 (1.3)]	3,208	10	7	10	[9.3 (0.8)]	3,184
	.7	10	[10 (0.0)]	14	10	[9.9 (0.3)]	6,040	10	8	10	[9.5 (0.5)]	6,148
AS2	.1	10	[10 (0.0)]	14	10	[8.7 (1.2)]	3,457	10	4	10	[9.1 (1.1)]	3,427
	.4	10	[10 (0.0)]	14	10	[9.4 (0.7)]	4,932	10	6	10	[9.2 (1.2)]	4,826
	.7	10	[8.7 (2.2)]	14	10	[7.7 (2.5)]	8,491	6	6	10	[7.8 (1.8)]	8,618
AS3	.1	5	[2.3 (3.2)]	13	5	[1.3 (2.2)]	8,772	4	3	4	[2.7 (3.1)]	6,953
	.4	4	[2 (3.0)]	13	7	[0.8 (0.6)]	8,317	1	5	7	[0.8 (0.6)]	7,149
	.7	5	[4 (4.8)]	14	7	[1.3 (1.2)]	8,423	2	5	8	[1.2 (1.1)]	7,725
AS4	.1	10	[10 (0.0)]	14	10	[9.7 (0.7)]	5,744	2	2	10	[9.7 (0.7)]	5,614
	.4	10	[10 (0.0)]	14	10	[8.6 (2.5)]	7,998	7	2	10	[8.7 (2.4)]	8,103
	.7	7	[2.6 (2.7)]	13	6	[0.7 (0.7)]	8,146	0	5	7	[0.8 (0.6)]	7,387

Note. Number of Global Maxima out of Ten Replications (G);
Average (and Standard Deviations) of Reported Global Maxima ([av.(sd.)]) and
Average Number of Iterations ($\bar{t}$) Over Ten Runs of the Algorithms and
Ten Replicated Data Matrices Having Moderate Discrimination Items

Table 4.8 shows that the algorithms with a random component (i.e., RAN1 & DET, RAN1 & RAN2 and SEQ & RAN2) performed best in finding global solutions for O_1 (i.e., the sums of #G were 120 for each algorithm). For O_2, RAN1 & RAN2 yielded the global solution 105 times, SEQ & RAN2 106 times, and RAN1 & DET 101 times. The completely deterministic algorithm performed worst for the two objective functions: it found the global solution 91 times for O_1 and 72 times for O_2. The values presented within brackets in Table 4.8 tell us that none of the algorithms yielded the global solutions every time it was run, but the random-move algorithms came closest. One may note that as ρ increased and as items loaded on more than one trait, the number of times a global solution was obtained decreased and the number of iterations needed to obtain a solution increased.

The relationship between the discrete partitionings and the objective function's value can explain the results of Table 4.8. The table shows that for low ρ and weak loadings on unintended traits (AS1 and AS2) all algorithms yielded the global solution. In these data matrices the relationship between the partitionings and the value of the objective function is relatively simple since there is only one maximum, and the fast deterministic algorithm can be used to find global maxima. For data having high ρ and high loadings on unintended traits (AS3), the relationship between partitionings and objective function is more complex because in addition to the global maximum, there are one or more local maxima, and the values of these maxima may be close to one another (also, see Tables 4.3 and 4.5). The complexity also means that the partitioning yielding these maxima are not very similar; that is, many items need to be moved before a different maximum is found. For this type of data, stochastic algorithms are needed. Table 4.8 further shows that functional relationship for O_1 was less complex than for O_2.

Overall, algorithms with a random move performed best. These algorithms, however, have the disadvantage that they require many iterations to converge. We tried variations of Equation 4.10 that reduced randomness (e.g., we used power t in stead of t/I) and made the algorithm faster. The algorithm converged earlier, but the algorithm's ability to find global solutions was reduced.

We conclude that the random-move algorithms (RAN1&RAN2 and SEQ&RAN2) should be preferred over all other algorithms, especially for O_2. However, the results also indicate that the use of RAN1&DET algorithms is defensible, especially for 'simple' data, since much speed is gained with little loss in accuracy.

4.6 Conclusions

Three new MSA approaches to resolve the optimization problems of sequential MSP were introduced in this chapter. The objective functions in these new methods incorporate reformulations of multidimensional Mokken scaling, and deterministic and stochastic non-hierarchical clustering algorithms were used to maximize these objective functions. In order to investigate the properties of the objective functions, we ignored the restrictions that are usually made in Mokken scaling analysis.

The first research question that we wanted to answer is: 'How successful are the three new objective functions in finding the underlying dimensionality of a data set?'. Objective function O_2 yielded the best results; O_1 performed somewhat better than the original sequential approach, and O_3 performed worse than the original sequential approach. Also, because the new methods using O_1 and O_2 found the true dimensionality they seem to be effective tools for this purpose, perhaps comparable to methods based on weak LI (e.g., Stout, 2002). This confirms that the H coefficient not only is a useful tool for scaling but also for dimensionality assessment.

The methods using O_1 and O_2 performed approximately equally well in most conditions of the study. This is not surprising because the two are strongly related. However, there are some differences. Objective function O_1 has the advantage that under $D = 1$ (i.e., unidimensionality) $O_1 = 0$ and, therefore, that deviation from unidimensionality can be determined. Another advantage of O_1 is that theoretically its value is maximized for $K = D$. Objective function O_2 does not have this advantage, but this can easily be remedied. A disadvantage of O_1 is that this objective function may not work well when D is large and when clusters are different (e.g., have unequal numbers of items or have differently discriminating items). The impact of these disadvantages requires further investigation. The interpretation of O_2 as the average within-cluster H_i is simpler than the interpretation of O_1. Both O_1 and O_2 use the H coefficient and, therefore, both join items on the basis of the slope of the IRFs. Based on the basis of the simulation study results and the properties presented above, O_2 is preferred to O_1.

The second research question was: 'Which algorithm should we use in the new MSA?'. A completely deterministic algorithm should clearly not be used, because it can only find global maxima for simple structure data with lowly correlated or uncorrelated latent traits. In general, the two stochastic-move algorithms (i.e., RAN1&RAN2 and SEQ&RAN2) performed best. For the preferred objective function O_2, however, the algorithm with the random start configuration and

the deterministic move (RAN1&DET) performed almost as well as the two best algorithms. Random-move algorithms increase the probability of obtaining global solutions, but it is a matter of taste if this increased precision justifies the large increase of computer labor. If high precision is required one should prefer the RAN1&RAN2 algorithm to the SEQ&RAN2. Then, it is not necessary to run an additional method (e.g., the sequential MSP method) in order to determine the initial configuration. It requires an extra action and it does not yield anything in return in terms of a higher likelihood of an optimal result (more optimal than global does not exist) or increased computational speed. When one recalls the overall procedure in which we investigate different values of K, the algorithm that finds a high enough precision solution at high speed is preferred: that is, the RAN1&DET algorithm.

4.7 Discussion

Some issues deserve further attention. First, when confronted with two highly correlated sets or sets with high loadings on unintended traits, researchers may differ in their opinion as to whether these sets should be joined. The NHCA methods can be applied whether or not a researcher prefers to join items or not, because the NHCA methods used in this chapter try to find the solution that maximizes the objective function for a given number of clusters. Thus, the decision about the number of clusters is left to the researcher. The presented methods provide various sources of information that help the researcher decide how many clusters to retain. Further research about this issue is needed.

Second, in MSA with the MSP software (Molenaar & Sijtsma, 2000), items will automatically satisfy Mokken scale conditions. We left the Mokken scale conditions out of the NHCA methods because we wanted to know whether the methods could be used to find globally optimal solutions and whether these solutions reflected the simulated dimensionality structure. If the Mokken scale conditions (i.e., with $c = 0.3$) were incorporated, weakly scalable items, for which the assignment of items into clusters is the most difficult, would have been left out of the analysis, and thus the limitations of the methods would have been difficult to investigate. Future research will address how the Mokken scale conditions can be incorporated into the new MSA methods.

Third, the IRF used for generating item response data has multiple inflection points. As a consequence, the degree of simple structure and the discrimination of the items is difficult to control. We redid some of the analysis with data generated with a M2-PLM (Van Abswoude et al., 2004) to ensure that the obtained results

were not an artifact of the model used for generating data. The results were similar to the results obtained with the mixture model.

Fourth, as discussed earlier, variations of the objective functions can be obtained by changing the definition of η. For example, in Equation 4.8 we used the average within-cluster H_i. One possible alternative objective function would maximize the average within-H_i and minimize the H_i between clusters, simultaneously. Using $\eta_i^k = -1$ when item i is not in cluster k would achieve this. This objective function does not have the convenient interpretation of the average within-cluster H_i, but it may be an appropriate objective function for determining the number of clusters. This is because the objective function would be reduced if items are in separate clusters when they should be in the same cluster. These issues will be addressed in future research.

Chapter 5

Scale Analysis Using Restricted Optimization Techniques

Abstract

Maximizing a function of H may be successful for selecting one or more unidimensional sets of items from a multidimensional test (see chapter 4). This method may compete with conditional covariance based methods in detecting the dimensionality of a data matrix. However, Mokken scale analysis aims to find sets of items (Mokken scales) that can be used to correctly order subjects on the basis of their underlying trait. Moreover, the Mokken scales obtained with Mokken scale analysis may not satisfy the local independence assumption of IRT models, which can indicate that the data is multidimensional. In this study, we offer a few suggestions how the alternative method introduced in chapter 4 may be extended such that Mokken scale conditions, conditions related to the monotone homogeneity model as well as other conditions can be satisfied in each of the obtained scales.

5.1 Introduction

As a follow-up of the stochastic non-hierarchical clustering algorithm (NHCA) methods that use the H coefficient for selecting items introduced in chapter 4 (also see Van Abswoude, Vermunt, & Hemker, 2003), we now present some issues that were ignored and give some suggestions how these issues can be resolved. For details about these NHCA methods, the sequential Mokken scaling method, and NIRT in general, we refer the reader to previous chapters of this thesis.

In chapter 4, the performance of NHCA methods as dimensionality assessment tools was investigated. The total number of clusters was considered to be known and using this preknowledge two out of the three investigated methods were successful in retrieving the underlying dimensionality of a data set. However, the resulting sets of items may not be Mokken scales because they do not satisfy the scaling conditions as proposed by Mokken (1971, p. 184).

In this paper, we propose a method to extend the NHCA methods such that Mokken scales can be created. In this method, the focus is redirected from a dimensionality assessment tool that uses a known number of clusters to a scaling tool. The user is given the choice to include other relevant conditions as well. These conditions can for example be related to the assumptions of the monotone homogeneity model (MHM) or the Double Monotonicity Model (DMM; e.g., Sijtsma & Junker, 1996; Sijtsma & Molenaar, 2002; Mokken & Lewis, 1982). Obviously, as more conditions are added to Mokken's conditions scaling becomes more restrictive than Mokken originally proposed. It also raises a number of problems that need to be solved. The extended method is referred to as 'MSA-E'; 'MSA' because the proposed minimal requirement is that for each obtained set of items (each scale) the Mokken (1971, p. 184) scale conditions should be satisfied, and 'E' (extended) because other relevant conditions may be chosen as well. The MSA-E is not a partitioning method in the strictest sense due to the possible presence of items that do not satisfy the various imposed conditions (denoted unscalable items). It is explained how a partitioning algorithm (like the stochastic NHCA) can be used for this problem. As in the usual MSA (and in the program MSP), the number of scales is considered to be unknown. With a few fictitious examples, we show that the suggested MSA-E method can be a useful tool for scaling.

5.2 Conditions in MSA-E

In the optimization problem we are now confronted with, we aim to maximize one objective function used in the stochastic NHCA methods, subject to a number

of conditions. The safest approach is to investigate every possible partitioning and calculate the value of the objective function. The partitioning that yields the highest value and satisfies the conditions is the partitioning solution that corresponds to the best possible division of items into Mokken scales. The number of possibilities, however, increases exponentially with the number of items and such an approach is therefore not regarded practical. We propose handling the relevant conditions using the RAN1&DET or RAN1&RAN2 algorithm presented in chapter 4. Other types of algorithms, for example based on linear programming (e.g., Danzig & Thapa, 1997), may require a different approach.

Some notation is needed to present the conditions that we propose for the new scaling methods. Let i denote a dichotomously scored item ($i = 1, \cdots, I$), and let (i, j) denote an item pair ($i \neq j$). We only discuss dichotomously scored items in this paper, but the extension of MSA-E to polytomous items seems to be straightforward. Let k denote a set of items forming a scale, and let K denote the total number of scales. In addition, let I^k denote the number of items selected in scale k, and let $\sum_{k=1}^{K} I^K$ denote the number of items selected in K scales. Let $\hat{H}$ be the sample estimate of the H coefficient (Loevinger, 1948; Mokken, 1971, p. 148).

Table 5.1: *Conditions Suggested for MSA-E*

Mokken scale:	S.1 $\widehat{cov}(X_i, X_j) > 0$
	S.2 $H_0 : H_i = 0$ is rejected in favor of $H_1 : H_i > 0$
	S.3 $\hat{H}_i \geq c$
MHM related:	S.1b $\hat{H}_{ij} > 0$
	S.4 based on weak LI
	S.5 based on nondecreasing IRFs
DMM related:	S.6 based on invariant item ordering
Other:	O.1 π_i-value
	O.2 reliability
	O.3 item content
	O.4 I^k
	...

Note. Conditions are proposed for all $i \in k$ and $i, j \in k$.

In Table 5.1, we suggest some conditions that might be relevant for creating scales. During item selection, except for one statistical test on the coefficients, only sample realizations are analyzed and for this reason Table 5.1 only includes sample

realizations. In addition, Table 5.1 does not offer an extensive and complete list, but rather gives the reader an impression of possible conditions that could be used in an MSA-E. The idea is that the user can choose between the different conditions based on his or her opinions about the appropriate scale for his or her application. For a Mokken scale, at least Conditions S.1−S.3 need to be satisfied. In addition to Conditions S.1−S.3, one may consider using one or more conditions that follow from the MHM assumptions (S.1b, S.4, and S.5) or from the DMM assumptions (S.1b, S.4, S.5 and S.6) when appropriate for a particular application. The resulting scale is a 'restricted' Mokken scale. One may note that the conditions related to the MHM or the DMM also can be used independent of the Mokken scale conditions. The resulting scale then satisfies some observable consequences of the MHM or DMM and need not be a Mokken scale. Also, one or more of the conditions O.1−O.4 can be used in combination with one of these types of scales. Details on the sequential item selection procedure (denoted sequential MSA) and the statistics can be found in chapter 4. For more information about the MSA conditions than presented below, the reader is referred to Mokken (1971), Molenaar and Sijtsma (2000), and Sijtsma and Molenaar (2002).

A few words of caution seem to be appropriate when defining new conditions because there are a number of things that can go wrong. Conditions may contradict each other or conditions may be too restrictive for the data such that a solution that satisfies the defined conditions does not exist (i.e., infeasibility problems). New conditions should only be introduced when there are good reasons for doing so. The intention is that users can define conditions in the input file that accompanies the software for MSA-E.

Mokken Scale Conditions

Condition $S.1$ is the sample equivalent of the condition that in earlier chapters was denoted Mokken Scale Condition 1. In the sequential MSA procedure (Mokken, 1971, pp. 191-193), S.1 should be satisfied for all item pairs that are joined into one scale. This condition requires that a scale consists of at least two items, because for one item covariances cannot be calculated and, thus, Condition S.1 cannot be satisfied. We propose that in MSA-E the condition is used in the same way.

Condition S.2 implies that $\hat{H}_i$ should be significantly larger than zero. One may note that for the item pair (i, j) that forms the start set of the scale in sequential MSA, Condition S.2 comes in a slightly different from; that is, $\hat{H}_{ij}$ should be significantly larger than zero. However, this is equivalent to S.2 because

for one item pair $\hat{H}_{ij} = \hat{H}_i = \hat{H}_j$.

Condition S.3 is the sample version of the condition that in earlier chapters was referred to as Mokken Scale Condition 2. In sequential MSA, Condition S.3 is defined for all i, and c is the lower bound for controlling the quality of items in a scale. Usually $c = 0.30$ is used, but in theory any value between 0 and 1 can be chosen (for rules of thumb, see Mokken, 1971, pp. 184-185).

Conditions S.1, S.2 and S.3 together are sample conditions for a Mokken scale and are, as a consequence, always used in MSA-E. The conditions that follow hereafter are new in the context of MSA.

MHM and DMM Related Conditions

Conditions can be based on observable consequences of the MHM or the DMM. Mokken scale condition 1 defines a Mokken scale and also is a necessary condition for the MHM, because under the MHM the responses between items have a nonnegative covariance (Mokken, 1971, p. 130-131, Theorem 1.4.1; also, see Holland & Rosenbaum, 1986). Thus, Mokken scale condition 1 is a necessary but not sufficient condition for the MHM to hold. One may note that Conditions S.1 and S.1b are equivalent.

For the remaining MHM and DMM related conditions, some problems need to be solved before they become realistic conditions in the optimization procedure. As an example, let us consider the LI assumption and let us rely on a existing method that has proven its worth: the DIMTEST-statistic (e.g, Stout et al., 2001). The first problem one is confronted with is what to do when weak LI is rejected. We would need a method that helps us determine which item yields the worst violation of weak LI and should consequently be rejected first. A method based on nonparametric estimates of IRFs is a possible choice to identify particular misfitting items and to find for which value of the latent trait these items depict misfit. Kim (1994), Douglas, Kim, Habing, and Gao (1998) and Habing (2001) used kernel smoothing to estimate covariance functions conditional on the latent trait to determine violations of LI as a function of the latent trait. One may note that incorporating Condition S.4 in each iteration step is computationally demanding since every time the scale-composition changes rest scores need to be determined and covariances estimated.

Other Conditions

In the MSA-E method, we try to provide an open structure that allows the user to specify a variety of conditions so that the method becomes useful for a wide

variety of applications. These conditions can be based on classical psychometric concepts like reliability (see O.2). Specifying conditions based on item content (see O.3) has the advantage that the user can obtain a test driven by the trait of choice (using $K = 1$, see chapter 4 and appendix) rather than the trait that is most dominant in the multidimensional data matrix. Item sets that should never be joined because they give clues with respect to each other's solution (enemy sets) can be chosen as scaling condition in a MSA-E. Some of these conditions have been used in computerized adaptive tests (CAT; e.g., Van der Linden, 1998; Veldkamp, 2001).

5.3 Handling the Conditions

A way to handle conditions is to include them in the objective function. A general way to do this is by means of a linear (or other type of) relationship between the objective function and the scaling conditions. This restricted objective function then has approximately the same interpretation as the original objective function. The restricted objective functions are corrected for the number of items in the scales. This reflects the fact that we prefer solutions with many scalable items to solutions with an even higher $\hat{H}$ but with fewer scalable items (see chapter 4.2). Given that we can find S scalable items, we prefer the clustering solution yielding the highest $\hat{H}$.

For the stochastic NHCA it is convenient to make a distinction between 'hard' an 'soft' conditions. The two types of conditions have in common that in the final clustering solution both should be satisfied. The distinction lies in the path towards the final solution that is allowed to be taken in the stochastic NHCA algorithm. Hard conditions need to be satisfied in every iteration step. In the terminology of genetic algorithms (e.g, Michalewicz, 1996), partitioning solutions that do not satisfy these conditions receive a death penalty. Thus, the outcome space is immediately restricted using hard conditions. For example, for Condition S.1 this means that only partitionings (or clustering solutions) for which $\hat{cov}(X_i, X_j) > 0$ for all $i, j \in k$ are possible for any iteration step.

Soft conditions are not necessarily satisfied at every iteration step. As one may recall, the stochastic NHCA algorithms were introduced in the new MSA methods to prevent local instead of global maximum values from occurring. In the stochastic-move algorithms (i.e., RAN1&RAN2 and SEQ&RAN2), this was achieved by temporarily allowing the objective function's value to decrease, and thus to allow solutions to deteriorate. The soft conditions may be violated at clustering solutions preceding the final solution but not at final solution. This is

done to be able to obtain near global solutions.

The choice between hard and soft conditions is more or less arbitrary in this implementation. Within this implementation, we choose to use S.1, S.2, and O.3 as hard conditions and the remaining as soft. Using the hard conditions we immediately restrict the outcome space similar to the sequential item selection procedure and the soft conditions allow us to find near global solutions.

Before the general formulation of the restricted objective functions is presented, first some additional notation is provided. Let O refer to any objective function defined in chapter 4, let O_R be an objective function that is restricted using one or more hard conditions chosen by the user, and using S.3 as a soft condition. Let $\zeta = 0$ denote that *all* hard conditions are satisfied; and $\zeta = -\infty$, otherwise. Items can be selected into a scale or a so-called dump cluster. Let $\mathcal{D}$ denote a dump cluster and let D refer to the number of items in $\mathcal{D}$. Because we introduce this new type of cluster, $\sum_{k=1}^{K} I^K \leq I$ ($I = \sum_{k=1}^{K} I^K + D$). For clarity η_i^k equals 1 when $i \in k$; and zero, otherwise (also, see the definitions of O_1, O_2 and O_3 in chapter 4). Condition $\nu_i^k = 1$ when $\hat{H}_i \geq c$ for $i \in k$; and $\nu_i^k = 0$, otherwise. Let S denote the number of items that satisfy the soft scaling condition; that is, $S = \sum_{k=1}^{K} \sum_{i=1}^{I} \nu_i^k$. For an explanation of the overall clustering procedure, the objective functions and notational conventions, see chapter 4.

The general shape of the restricted objective functions is defined as follows:

$$O_R = O + \zeta + S. \tag{5.1}$$

Objective function O_R aims to achieve to following three purposes. First, solutions where hard conditions are not satisfied need to be excluded from the solution space; this is achieved by ζ. This is because if one of the hard conditions is not satisfied for a partitioning, $O_R = -\infty$, and thus the probability that this partitioning is selected equals zero (see Equation 4.10 in chapter 4). The second purpose is to scale as many items as possible. This is achieved by the inclusion of S. Third, if there are unscalable items, the value of the objective function should be higher for partitioning solutions where these items are in the dump cluster rather than in one of the scales. One may note that this general function O_R is presented to explain the logic behind the restricted objective functions, but that the specific implementation for each objective function is different.

The objective functions presented in chapter 4 were not equally promising. The first objective function O_1 using $c^* = \bar{H}_{ij}$ has the advantage that the $\hat{H}_{ij}$, matrix can be used directly and that, except for the calculation of the objective function itself, no further calculations are necessary. This property makes the

stochastic NHCA method conveniently fast. Another advantage is that O_1 is typically maximized at the appropriate number of scales as long as the correlations between latent traits are not too high (see chap. 4 for details). Both splitting an item pair that is driven by one trait and joining two items that are driven by different traits yield a negative contribution to the objective function. Thus, when either too few or too many scales are specified in a partitioning, the objective function will be lower than the globally optimal value. An important drawback of O_1 is, however, that it requires approximately equal Hs for each scale (i.e., equal numbers of items, equal discrimination) and that O_1 may perform less well as the number of traits underlying the data increases. The effect of these requirements needs more extensive investigation. For the moment, we do not consider O_1 in an restricted format because the other objective functions were more promising.

Objective function O_2 aims to maximize the scalability of each item's H_i. An important advantage of O_2 is that given that K scales are investigated, this objective function was most successful in assessing dimensionality. A disadvantage of O_2 is that it is not very likely that it is maximized at the correct number of scales, since O_2 increases as K increases. The following restricted objective function O_{2R} may be useful:

$$O_{2R} = (\sum_{k=1}^{K} I^K)^{-1} \sum_{k=1}^{K} \sum_{i=1}^{I} \eta_i^k \hat{H}_i^k + \zeta + \sum_{k=1}^{K} \sum_{i=1}^{I} \nu_i^k. \tag{5.2}$$

One may easily recognize the general form of Equation 5.1 in O_{2R}. Some other details are worth pointing out. One may note that a partitioning that is evaluated can have *scalable* and *unscalable* items in each scale k. The goal of a method using O_{2R} is to find a partitioning that maximizes the H_i of the *scalable* items. The convenient interpretation of O_{2R} as the average within-scale H_i of the items in each scale can be obtained by simply subtracting the number of scalable items from O_{2R}. The multiplication with $(\sum_{k=1}^{K} I^K)^{-1}$ not only aids the interpretation of the objective function as in O_2, but also aims at achieving that O_{2R} is higher for a partitioning that assigns an *unscalable* item to $\mathcal{D}$ instead of a scale k. The value O_{2R} also is expected to become higher when an unscalable item is moved to $\mathcal{D}$ because the average H_i of the remaining items will increase when this item is moved to $\mathcal{D}$.

As an illustration of O_{2R}, consider the following hypothetical example using six items driven by two uncorrelated latent traits θ_1 and θ_2. Item1, Item2 and Item3 have high discriminations with respect to θ_1 and Item4 and Item5 have high discriminations with respect to θ_2. Item3 and Item6 are weakly driven by θ_2 and Item4 is weakly driven by θ_1. Table 5.2 shows the effects of moving one item to a

Table 5.2: *Value of O_{2R} when Moving One Item to a Scale (Scale1 or Scale2) or the Dump Cluster ($\mathcal{D}$) from a Start Partitioning (underlined)*

	$K=1$		$K=2$		
	Scale1	$\mathcal{D}$	Scale1	Scale2	$\mathcal{D}$
Item1	<u>3.249</u>	1.208	<u>5.553</u>	4.348	4.481
Item2	<u>3.249</u>	0.215	<u>5.553</u>	3.409	4.563
Item3	<u>3.249</u>	0.215	<u>5.553</u>	4.395	4.522
Item4	<u>3.249</u>	3.272	3.257	<u>5.553</u>	3.439
Item5	<u>3.249</u>	3.290	3.284	<u>5.553</u>	3.487
Item6	<u>3.249</u>	3.305	5.552	<u>5.553</u>	5.772

scale k or cluster $\mathcal{D}$ on O_{2R} for $K = 1$ and $K = 2$. We start out with a particular partitioning $\mathcal{P}$, which is underlined in Table 5.2. We used $c = 0.3$, throughout. Negative $\hat{H}_{ij}$s did not occur.

For the 1-scale partitioning, we started out with all items in one scale, yielding the following item H_is: $H_1 = .34$, $H_2 = .31$, $H_3 = .34$, $H_4 = .23$, $H_5 = .19$ and $H_6 = .09$[1]. The start configuration has $O_{2R} = 3.249$, which means that three out of six items were scalable and the mean $\hat{H}_i$ of the scaled items was .249. Moving Item6 to $\mathcal{D}$ yields the largest increase in O_{2R}. Which item is actually moved depends on the type of algorithm the method uses. Regardless of which algorithm is used, however, one move is not sufficient for attaining convergence. This is because with three scalable items in Scale1 and one unscalable item in $\mathcal{D}$, there still are two unscalable items left in Scale1. Moving both of these unscalable items to $\mathcal{D}$ may yield an higher value for the objective function O_{2R} than 3.305. For $K = 2$ we start out with the simulated dimensionality. Scale1 had item scalability values of $H_1 = .77$, $H_2 = .69$ and $H_3 = .71$, and Scale2 had item scalability values of $H_4 = .58$, $H_5 = .50$ and $H_6 = .06$. Table 5.2 shows the best move is moving Item6 to $\mathcal{D}$. For this small example O_{2R} seems to do what it is supposed to do. Increasing K does not yield a higher number of scalable items and, therefore, we stop at $K = 2$.

The third objective function O_3 aims at maximizing the average scale H. Compared to O_1 and O_2, objective function O_3 was most directly related to what the

[1]We used the stochastic NHCA software for calculating these values. This package gives two decimal places for the item's H_i (as does package MSP). To give the opportunity to verify O_{2R} we also provide the values with three decimal places: $H_1 = .342$, $H_2 = .305$, $H_3 = .337$, $H_4 = .228$, $H_5 = .194$ and $H_6 = .089$ (only for this example).

Table 5.3: *Value of O_{3R} when Moving One Item to a Scale (Scale1 or Scale2) or the Dump Cluster ($\mathcal{D}$) from a Start Partitioning (underlined)*

Items	Scale1	Scale2	$\mathcal{D}$
Item1	<u>4.520</u>	$-\infty$	$-\infty$
Item2	5.580	<u>4.520</u>	4.637
Item3	<u>4.520</u>	$-\infty$	$-\infty$
Item4	2.158	<u>4.520</u>	2.447
Item5	2.175	<u>4.520</u>	2.465
Item6	4.304	<u>4.520</u>	4.537

sequential clustering procedure aims to achieve. Objective function O_3 has a few weaknesses as well. Similarly to O_2, O_3 cannot be used for determining the number of scales. Moreover, the unrestricted objective function was not very successful as a tool for dimensionality assessment (Van Abswoude et al., 2003). For $K = 2$, O_3 typically yielded a partitioning with one cluster consisting of the item pair with the highest $\hat{H}_{ij}$, and the other cluster with the remaining items. This was because the reduction in H caused by the larger cluster could not compensate for the increase in H caused by the smaller cluster which, compared to the other cluster, had a very high H. Adding scaling conditions to the optimization problem as in Equation 5.1 may be sufficient to resolve this drawback. The third objective function can be adjusted in a similar way as O_{2R}:

$$O_{3R} = K^{-1} \sum_{k=1}^{K} \hat{H}^k + \zeta + \sum_{k=1}^{K} \sum_{i=1}^{I} \nu_i^k. \tag{5.3}$$

For our small example, Table 5.3 shows the value of O_{3R} when one item is moved from a start configuration. As start configuration we use the typical result obtained with the unrestricted stochastic NHCA method: the best pair (Item1, Item3) in Scale1 and the remaining items in Scale2.

In Table 5.3, $O_{3R} = -\infty$ was obtained when Item1 or Item3 is moved out of Scale1, because for one item $cov(X_i, X_j)$ is undefined. Moving Item2 to Scale1 yields the largest improvement. Table 5.3 shows that O_{3R} can yield the simulated dimensionality.

5.4 Determining the Number of Scales

How to determine the appropriate number of scales was discussed extensively in chapter 4 (see chapter 4 and appendix for details). As was explained there, different sources of information, such as substantive information, the value of the objective function, and the number of scalable items should be consulted to determine the appropriate number of scales. Surely the practitioner should not continue increasing the number of scales when the maximum number of scalable items has been reached.

The scree plots used in chapter 2 may not be very informative for determining the appropriate number of scales in an MSA-E. Due to the extension of MSA with several conditions and a dump cluster, the value of the objective function may not show a predictable pattern as the number of investigated scales increases. This is because dump-cluster items do not contribute to the objective function's value, and scalable items do. The value of the objective function may therefore for $K = 1$ be as high as for $K = 3$. Thus, investigating scree plots is not a tenable method on which to base a decision about the number of scales.

The question about the number of clusters might be tackled by other means. In particular, it might be possible to change the definition of η such that the objective function can be maximized at the appropriate K (also, see O_1). The alternative definition of η equals: $\eta_i^k = 1$ if $i \in k$; $\eta_i^k = 0$ if $i \in \mathcal{D}$; and $\eta_i^k = -1$, otherwise. This alternative η can be introduced in O_{2R} and in O_{3R} (if it is written as the weighted sum of the H_is). The convenient interpretations of the objective functions as the average H_i (O_{2R}) and the average H (O_{3R}) is lost, however. The consequences of changing the definition of η on dimensionality assessment and scaling requires further study.

5.5 Final Remarks

The purpose of this chapter was to discuss some unresolved problems of the alternative clustering method presented in chapter 4. We provided some ideas for solving these problems. Preliminary results seem promising, although it is also clear that further study is needed.

Future research may be aimed at the completion of the MSA-E methods and may investigate the effect of different types of data on yielding different types of (Mokken) scales. For this purpose, we may simulate violations of the nonparametric IRT assumptions in addition to nonparametric IRT-conform data and investigate if MSA-E can detect the misfit. Next, we may investigate the effect of

generating multiple scales with different scale H values on yielding different types of scales. Lastly, we may investigate the stability of the results. Previous research showed that the objective functions are to a large degree stable in repeated sampling (Van Abswoude et al., 2003) but the stability may be reduced when scaling conditions are added.

Appendix

Stochastic and deterministic methods using Mokken's H coefficient were introduced in chapter 4, and to these methods the Mokken scale conditions were added in chapter 5. The number of clusters in these methods were regarded to be known in advance. This appendix addresses how these methods handle an unknown number of clusters and how the researcher may choose between solutions that have a different number of clusters. The same notation as in chapters 4 and 5 is used.

The deterministic and stochastic NHCA methods including the Mokken scale conditions have the following three basic steps:

1 the user chooses c (i.e., Mokken scale condition 2) and the maximum number of clusters that are investigated, $K^*(K = 1, 2, \cdots, K^*)$;

2 the program searches for optimal solutions for $K = 1, \cdots, K^*$;

3 the user chooses a different value for c or K^*, or chooses the final solution using all available sources of information.

Now, each step is explained in more detail. First, the researcher should decide on what the desired scale strength is, given his/her test application. This means that the user chooses the value of c and maybe the value of other scale restricting conditions (see chapter 5). The desired scale strength is also chosen in advance in the sequential method. At this stage (this is new), the user may also choose the upper bound of the number of clusters that are investigated, K^*. This is optional and one should not be concerned when he/she finds it difficult to find a suitable K^*. When the upper bound is not chosen all possible values of K are investigated. Otherwise, one could use the number of latent variables one expects on the basis of substantive knowledge and add two to this number for certainty. This is because when the practitioner is certain about theoretical basis of his or her test, the test may be driven by other, unintended, traits. For example, contextual math problems may be sensitive to math and language skills. When the practitioner

intends to create one single test, $K = 1$ can be chosen. An extra cluster that need not be included in K^*, and which may be empty, is the so called 'dump' cluster. Items that do not satisfy the scale conditions (unscalable items) are entered in this cluster (for more details see chapter 5).

Next, O_1, O_2 or O_3 is maximized for $K = 1, \cdots, K^*$. This yields K^* optimal or nearly optimal partitionings.

Finally, the user can choose which of the K^* optimized solutions best reflects the dimensionality or scalability of his or her test data. This choice can be based one or more sources of information; these are: substantive knowledge; the purpose of scaling; the values of the maximized objective functions; the item and scale H values; the number of scalable items; and maybe other sources of information. The practitioner may decide which he/she prefers. On the basis of this information, one may also decide to try out different c or K^*.

User-defined constant c plays a slightly different in the new MSA methods. It plays the same role in rejecting items from each of the scales that do not satisfy the Mokken scale conditions. In the sequential method, c also plays a role in yielding a particular number of scales. This indirect effect of c on the number of scales is no longer present in the new method. Regardless of whether the practitioner prefers the 2-, 3-, or 4-cluster solution, using the new MSA methods each obtained cluster satisfies the scale conditions chosen in step 1.

References

Ackerman, T. (1996). Graphical representation of multidimensional item response theory. *Applied Psychological Measurement, 20*, 311-329.

Bacon, D. R. (2001). An evaluation of cluster analytic approaches to initial model specification. *Structural Equation Modelling, 8 (3)*, 379-429.

Becht, M. C., Poortinga, Y. H., & Vingerhoets, A. J. J. M. (2001). Crying across countries. In A. J. J. M. Vingerhoets & R. R. Cornelius (Eds.) *Adult crying: A biopsychosocial approach* (pp. 135-158). Hove, UK: Brunner-Routledge.

Béguin, A. A., & Glas, C. A. W. (2001). MCMC estimation and some model-fit analysis of multidimensional IRT models. *Psychometrika, 66*, 541-562.

Berthold, M., & Hand, D. J. (1999). *Intelligent data analysis: an introduction.* New York: Springer.

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. M. Lord & M. R. Novick, *Statistical theories of mental test scores* (pp. 397-479). Reading, MA: Addison-Wesley.

Cronbach, L. J. (1990). *Essentials of psychological testing (5th ed.).* New York: Harper & Row.

Danzig, G. B., & Thapa, M. N. (1997). *Linear programming.* New York: Springer.

De Groot, A. D. (1961). *Methodologie: Grondslagen van onderzoek en denken in de gedragswetenschappen.* 's Gravenhagen, The Netherlands: Mouton.

Douglas, J., & Cohen, A. (2001). Nonparametric item response function estimation for assessing parametric model fit. *Journal of Educational Measurement, 25*, 234-243.

Douglas, J., Kim, H. R., Habing, B., & Gao, F. (1998). Investigating local dependence with conditional covariance functions. *Journal of Educational Measurement, 23*, 129-151.

Douglas, J., Kim, H. R., Roussos, L., Stout, W. F., & Zhang, J. (1999). *LSAT dimensionality analysis for the December 1991, June 1992, and October 1992 administrations.* Newton, PA: LSAT.

Emons, W. H. M. (2003). *Detection and diagnosis of misfitting item-score patterns.* Unpublished doctoral dissertation, Tilburg University, The Netherlands,.

Everitt, B. S., Landau, S., & Leese, M. (2001). *Cluster analysis.* London/New York: Oxford University Press Inc.

Fischer, G. H., & Molenaar, I. W. (1995). *Rasch models: Foundations, recent developments and applications.* New York: Springer.

Glas, C. A. W., & Verhelst, N. D. (1995). Testing the Rasch model. In G.H. Fischer & I.W. Molenaar (Eds.), *Rasch models. Foundations, recent developments, and applications* (pp. 69-95). New York: Springer.

Grayson, D. A. (1988). Two-group classification in latent trait theory: Scores with monotone likelihood ratio. *Psychometrika, 53*, 383-392.

Guttman, L. (1950). The basis for scalogram analysis. In S.A. Stouffer, L. Guttman, E.A. Suchman, P.F. Lazersfeld, S.A. Star, & J.A. Clausen (Eds.), *Measurement and prediction* (pp. 60-90). Princeton, NJ: Princeton University Press.

Habing, B. (2001). Nonparametric regression and the parametric bootstrap for local dependence assessment. *Applied Psychological Measurement, 25*, 221-233.

Hambleton, R. K., & Swaminathan, H. (1985). *Item response theory: Principles and applications.* Dordrecht, The Netherlands: Kluwer-Nijhoff Publishing.

Hattie, J., Krakowski, K., Rogers, H. J., & Swaminathan, H. (1996). An assessment of Stout's index of essential unidimensionality. *Applied Psychological Measurement, 20*, 1-14.

Hemker, B. T., Sijtsma, K., & Molenaar, I. W. (1995). Selection of unidimensional scales from a multidimensional item bank in the polytomous Mokken IRT model. *Applied Psychological Measurement, 19*, 337-352.

Hemker, B. T., Sijtsma, K., Molenaar, I. W., & Junker, B. W. (1997). Stochastic ordering using the latent and the sum score in polytomous IRT models. *Psychometrika, 62*, 331-347.

Holland, P. W., & Rosenbaum, P. R. (1986). Conditional association and unidimensionality in monotone latent variable models. *The Annals of Statistics, 14*, 1523-1543.

Hunter, J. E. (1973). Methods of ordering the correlation matrix to facilitate visual inspection and preliminary cluster analysis. *Journal of Educational Measurement, 10*, 51-61.

Ip, E. H. (2001). Testing for local dependency in dichotomous and polytomous item response models. *Psychometrika, 66*, 109-132.

Junker, B. W. (1993). Conditional association, essential independence and monotone unidimensional item response models. *The Annals of Statistics, 21*, 1359-1378.

Kelderman, H., & Rijkes, C. P. M. (1994). Loglinear mutlidimensional IRT models for polytomously scored items. *Psychometrika, 59*, 149-176.

Kim, H. R. (1994). *New techniques for the dimensionality assessment of standardized test data.* Unpublished doctoral dissertation, University of Illinois, Urbana-Champaign.

Knol, D. L., & Berger, M. P. F. (1991). Empirical comparison between factor analysis and multidimensional item response models. *Multivariate Behavioral Research, 26*, 457-477.

Loevinger, J. (1948). The technique of homogenous tests compared with some aspects of "scale analysis" and factor analysis. *Psychological Bulletin, 45*, 507-530.

Lord, F. M., & Novick, M. R. (1968). *Statistical theories of mental test scores.* Reading, MA: Addison-Wesley.

McDonald, R. P. (1982). Linear versus nonlinear models in item response theory. *Applied Psychological Measurement, 6*, 379-396.

McDonald, R. P. (1985). *Factor analysis and related models.* New Jersey: Hillsdale.

McDonald, R. P. (2000). A basis for multidimensional item response theory. *Applied Psychological Measurement, 24*, 99-114.

Meijer, R. R. (1994). *Nonparametric person fit analysis.* Unpublished doctoral thesis, Vrije Universiteit, Amsterdam.

Michalewicz, Z. (1996). *Genetic algorithms + data structures = evolution programs.* New York: Springer.

Miecskowski, T. A., Sweeney, J. A., Haas, G., Junker, B. W., Brown, R. P., & Mann, J. (1993). Factor composition of the suicide intent scale. *Suicide and Life Threatening Behavior, 23*, 37-45.

Mokken, R. J. (1971). *A theory and procedure of scale analysis.* Berlin: De Gruyter.

Mokken, R. J., & Lewis, C. (1982). A nonparametric approach to the analysis of dichotomous item responses. *Applied Psychological Measurement, 6*, 417-430.

Mokken, R. J., Lewis, C., & Sijtsma, K. (1986). Rejoinder to 'the Mokken scale: A critical discussion'. *Applied Psychological Measurement, 10*, 279-285.

Molenaar, I. W. (1982). Mokken scaling revisited. *Kwantitatieve Methoden, 3(8),* 145-164.

Molenaar, I. W. (1983). Some improved diagnostics for failure of the Rasch model. *Psychometrika, 48,* 49-72.

Molenaar, I. W. (1991). A weighted loevinger H-coefficient extending the mokken scaling to multicategory items. *Kwantitatieve Methoden, 12(37),* 97-117.

Molenaar, I. W., & Sijtsma, K. (2000). Users manual MSP5 for Windows. A program for Mokken scale analysis for polytomous items [Software manual]. Groningen, The Netherlands: iec ProGAMMA.

Nandakumar, R., & Stout, W. F. (1993). Refinements of Stout's procedure for assessing latent trait unidimensionality. *Journal of Educational Statistics, 18,* 41-68.

Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests.* Copenhagen: Danish Institute for Educational Research.

Reckase, M. D. (1997). A linear logistic multidimensional model for dichotomous item response data. In W.J. van der Linden & R.K. Hambleton (Eds.), *Handbook of modern item response theory,* (pp. 271-286). New York: Springer.

Reckase, M. D., & McKinley, R. L. (1991). The discriminating power of items that measure more than one dimension. *Applied Psychological Measurement, 15,* 361-373.

Reveille, W. (1979). Hierarchical cluster analysis and the internal structure of tests. *Multivariate Behavioral Research, 14,* 57-74.

Roussos, L., & Ozbek, O. (2003). Formulation of the DETECT population parameter and evaluation of DETECT estimator bias. *Paper presented at the annual meeting of the National Council on Measurement in Education, Chicago, april.*

Roussos, L. A. (1992). *Hierarchical agglomerative clustering computer program user's manual.* Unpublished doctoral dissertation, University of Illinois, Urbana-Champaign.

Roussos, L. A., Stout, W. F., & Marden, J. I. (1998). Using new proximity measures with hierarchical cluster analysis to detect multidimensionality. *Journal of Educational Measurement, 35,* 1-30.

Samejima, F. (1995). Acceleration model in the heterogeneous case of the general graded response model. *Psychometrika, 60,* 549-572.

Samejima, F. (2000). Logistic positive exponent family of models: Virtue of assymmetric item characteristic curves. *Psychometrika, 65,* 319-335.

Scheirs, J. G. M., & Sijtsma, K. (2001). The study of crying: Some methodological considerations and a comparison of methods for analyzing questionnaires. In J. J. M. Vingerhoets & R. R. Cornelius (Eds.) *Adult crying: A biopsychosocial approach* (pp. 277-298). Hove, UK: Brunner-Routledge.

Schweizer, K. (1991). Classifying variables on the basis of disaggregate correlations. *Multivariate Behavioral Research, 26*, 435-455.

Sijtsma, K. (1998). Methodology review: Nonparametric IRT approaches to the analysis of dichotomous item scores. *Applied Psychological Measurement, 22*, 3-31.

Sijtsma, K., & Junker, B. W. (1996). A survey of theory and methods of invariant item ordering. *British Journal of Mathematical and Statistical Psychology, 49*, 79-105.

Sijtsma, K., & Molenaar, I. W. (2002). *Introduction to nonparametric item response theory.* Thousand Oaks, CA: Sage.

Sijtsma, K., & Van der Ark, L. A. (2001). Progress in NIRT analysis of polytomous item scores: dilemmas and practical solutions. In A. Boomsma, M. A. J. van Duijn, & T. A. B. Snijders (Eds.), *Essays on item response theory* (pp. 297-318). New York: Springer.

Sokal, R. R., & Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin, 38*, 1409-1438.

SPSS Inc. (1998). *SPSSX user's guide.* New York: McGraw-Hill.

Stout, W. F. (1987). A nonparametric approach for assessing latent trait unidimensionality. *Psychometrika, 52*, 293-325.

Stout, W. F. (1990). A new item response theory modeling approach with applications to unidimensionality assessment and ability estimation. *Psychometrika, 55*, 293-325.

Stout, W. F. (2002). Psychometrics: from practise to theory and back: 15 years of nonparametric multidimensional IRT, DIF/test equity, and skills diagnostic assessment. *Psychometrika, 67*, 485-518.

Stout, W. F., Douglas, J., Junker, B. W., & Roussos, L. (1993). *DIMTEST manual.* Unpublished manuscript, University of Illinois, Urbana-Champaign.

Stout, W. F., Goodwin Froelich, A., & Gao, F. (2001). Using resampling methods to produce an improved DIMTEST procedure. In A. Boomsma, M.A.J. van Duijn, & T.A.B. Snijders (Eds.), *Essays on item response theory* (pp. 357-375). New York: Springer.

Stout, W. F., Habing, B., Douglas, J., Kim, H. R., Roussos, L. A., & Zhang, J. (1996). Conditional covariance-based nonparametric multidimensionality assessment. *Applied Psychological Measurement, 20*, 331-354.

Swingler, K. (1996). *Applying neural networks: A practical guide.* London: Academic Press.

Thorndike, R. L., Hagen, E. P., & Sattler, J. M. (1986). *Stanford-Binet intelligence scale, (4th ed.).* Chicago: Riverside Publishing.

Van Abswoude, A. A. H., Van der Ark, L. A., & Sijtsma, K. (2004). A comparative study on test dimensionality procedures under nonparametric IRT models. *Applied Psychological Measurement, 28(1)*, 1-23.

Van Abswoude, A. A. H., & Vermunt, J. K. (2003). Some alternative clustering methods for Mokken scale analysis. In H. Yanai, A. Okada, Y. Kado, J.J. Meulman (Eds.), *New developments in psychometrics* (pp. 625-630). Tokyo: Springer.

Van Abswoude, A. A. H., Vermunt, J. K., & Hemker, B. T. (2003). Assessing dimensionality by maximizing H coefficient based objective functions. *Manuscript submitted for publication.*

Van Abswoude, A. A. H., Vermunt, J. K., Hemker, B. T., & Van der Ark, L. A. (in press). Mokken scale analysis using hierarchical clustering procedures. *Applied Psychological Measurement.*

Van der Linden, W. J. (1998). Assembling tests for the measurement of multiple traits. *Applied Psychological Measurement, 20*, 373-388.

Veldkamp, B. P. (2001). *Principles and methods of constrained test assembly.* Unpublished doctoral dissertation, University of Twente, The Netherlands.

Vermunt, J. K. (1997). *LEM: A general program for the analysis of categorical data.* Unpublished doctoral dissertation, Tilburg University, The Netherlands.

Zhang, J. (1996). *Some fundamental issues in item response theory with applications.* Unpublished doctoral dissertation, University of Illinois, Urbana-Champaign.

Zhang, J., & Stout, W. F. (1999a). Conditional covariance structure of generalized compensatory multidimensional items. *Psychometrika, 64*, 129-152.

Zhang, J., & Stout, W. F. (1999b). The theoretical DETECT index of dimensionality and its application to approximate simple structure. *Psychometrika, 64*, 213-249.

Zhang, Y. O., Yu, F., & Nandakumar, R. (2003). The impact of conditional scores on the performance of DETECT. *Paper presented at the annual meeting of the National Council on Measurement in Education, Chicago, april.*

Summary

The aim of this thesis was to investigate how we can successfully select one or more sets of items driven by a single latent trait from a test driven by multiple latent traits. This method should use the relatively weak assumptions from nonparametric item response theory (IRT). Compared to parametric IRT models, nonparametric models frequently show a better fit to test data. This renders nonparametric IRT models more appropriate when ordinal rather than interval measurement is sufficient for the measurement purpose at hand. In the introduction to the chapters an illustration of the relevance of multidimensionality assessment and possible approaches were presented.

Chapter 1 presented two models on which dimensionality assessment methods in nonparametric IRT can be based: essentially and strictly unidimensional models. The three methods DETECT, DIMTEST and HCA/CCPROX are based on the essentially unidimensional model and use covariances conditional on the latent trait to assess dimensionality. The method MSP is based on the strictly unidimensional model and uses the combination of the H coefficient and scale quality conditions. In a simulation study, the four methods were compared with respect to their ability to retrieve a simulated dimensionality structure. In general, it was found that the conditional covariance-based methods were often better in retrieving the dimensionality structure of the data than the H coefficient-based method.

In Chapter 2, four hierarchical alternatives for the item selection algorithm used for Mokken Scale Analysis (MSA) were proposed. Attractive properties of the methods that use these different algorithms were the simplicity and the standard availability of three of these methods in SPSS (SPSS, 1998). A fourth could not be reproduced with SPSS, but was especially programmed for MSA. Another attractive property is that the methods provide the opportunity to investigate the process by which sets of items are joined. By means of a simulation study and an empirical example, it was shown that the complete linkage method and

the scale linkage method with the H coefficient as proximity metric performed better than MSA's original selection method and the other hierarchical methods in dimensionality assessment.

The third chapter discussed the effects that clustering algorithms may have on finding the underlying dimensionality of data. Using simulated examples, for different algorithms (including hierarchical algorithms) it was shown where in the process of clustering items things might go wrong in the sense that suboptimal solutions may be found and, consequently, that the underlying dimensionality may not be retrieved.

The next chapter, Chapter 4, introduces three alternative methods with the aim to reduce the probability of obtaining suboptimal solutions. These methods used deterministic and stochastic versions of non-hierarchical clustering (NHCA) algorithms and clearly defined scaling objectives in both unidimensional and multidimensional contexts. Specific scaling conditions were not included. Using a simulation study, it was shown that stochastic NHCA algorithms may be used for obtaining optimal (or, near optimal) solutions. Moreover, these NHCA methods based on the H coefficient seem to be able to compete with the conditional covariance-based methods in yielding sets that reflect the underlying dimensionality of data.

Finally, in Chapter 5, suggestions were given on how the new NHCA methods discussed in Chapter 4 may be extended so that they become useful for creating multiple Mokken scales; that is, incorporating the MSA conditions. The chapter also explained how other interesting conditions may be imposed on the data as well.

Samenvatting (Summary in Dutch)

Het doel van dit proefschrift was het achterhalen hoe we het beste een of meer verzamelingen eendimensionale items uit een meerdimensionale test kunnen selecteren. Deze methode zou gebruik moeten maken van de relatief zwakke aannamen die binnen de nonparametrische item respons theorie (IRT) gelden. In vergelijking met methoden die gebaseerd zijn op parametrische IRT zal deze methode voor meer onderzoeksgegevens een goede passing met het gebruikte model moeten opleveren. Dit maakt de methode geschikt wanneer een ordinaal meetniveau voldoende is voor het beoogde onderzoeksdoel. In de inleidende pagina's van het proefschrift wordt uitgelegd waarom het onderzoeken van de dimensionaliteit van onderzoeksgegevens belangrijk is en worden mogelijke benaderingen gepresenteerd.

Hoofdstuk 1 beschrijft twee nonparametrische IRT modellen waarop methoden voor dimensionaliteitsonderzoek gebaseerd kunnen zijn: het essentiële en het strikte unidimensionale model. De methoden DETECT, DIMTEST en HCA/CCPROX zijn gebaseerd op het essentiële unidimensionale model en maken gebruik van covarianties conditioneel op de latent trek om de dimensionaliteit van data vast te stellen. De methode MSP is gebaseerd op het strikte unidimensionale model en maakt gebruik van de H coëfficiënt en schalingscondities. Door middel van een simulatiestudie werden de vier methoden met elkaar vergeleken op de mate waarmee ze ons in staat stellen de dimensionaliteit van test data terug te vinden. De conditionele covariantie-gebaseerde methoden bleken het op dit aspect beter te doen dan de op de H coëfficiënt gebaseerde methode.

In Hoofdstuk 2 worden vier hiërarchische alternatieven voor het item selectie algoritme van Mokken-schaal analyse (MSA) voorgesteld. Aantrekkelijke eigenschappen van deze methoden zijn de eenvoud en het standaard aanwezig zijn van drie van deze methoden in SPSS (SPSS, 1998). Een vierde kon niet met SPSS gereproduceerd worden en werd speciaal voor deze toepassing geprogrammeerd. Een andere aantrekkelijke eigenschap was de mogelijkheid die deze methoden bieden om het clusteringsproces te onderzoeken. Met behulp van een simulatiestudie en een empirisch voorbeeld lieten we zien dat de "complete linkage" methode en de

"schaal linkage" methode in combinatie met de H coëfficiënt als afstandsmaat het beter doen dan MSA's originele itemselectiemethode en de andere onderzochte hiërarchische methoden.

Het derde hoofdstuk bespreekt de effecten die cluster algoritmes kunnen hebben op het vinden van de dimensionaliteit. Gebruik makend van gesimuleerde voorbeelden lieten we voor diverse algoritmes zien op welk moment in het clusterproces verkeerde stappen genomen kunnen worden waardoor de optimale oplossing niet gevonden wordt. Een gevolg hiervan kan zijn dat de dimensionaliteit niet achterhaald wordt.

In het volgende hoofdstuk, Hoofdstuk 4, introduceerden we drie alternatieve methoden met het doel de kans op suboptimale oplossingen te reduceren. Binnen deze methoden werden deterministische en stochastische varianten van een niet-hiërarchisch cluster (NHCA) algoritme gebruikt. Ook werden doelfuncties ontwikkeld waarin voor zowel het eendimensionale als het meerdimensionale geval eenduidig werd geformuleerd wat een optimaal clusterresultaat is. Specifieke schalingscondities werden achterwege gelaten. Door middel van een simulatiestudie lieten we zien dat stochastische NHCA algoritmes gebruikt kunnen worden om optimale (of bijna optimale) oplossingen te verkrijgen. Bovendien bleken de op de H coëfficiënt gebaseerde methoden, toen we deze stochastische algoritmes toepasten, de dimensionaliteit van de gegevens ongeveer even goed terug te vinden als de op conditionele covarianties gebaseerde methoden.

Ten slotte werden in Hoofdstuk 5 suggesties gegeven hoe de nieuwe methoden die we in Hoofdstuk 4 introduceerden uitgebreid kunnen worden zodat ze bruikbaar worden om meerdere Mokkenschalen te gelijkertijd te ontwikkelen. Daarnaast werd uitgelegd hoe met deze methoden andere interessante condities aan de onderzoeksgegevens opgelegd kunnen worden.

ISBN 90-9018047-8